Université de Liège
Faculté des Sciences Appliquées
Département d'Aérospatiale, Mécanique et Matériaux
Vibrations et Identification des Structures

Integrated Simulation and Reduced-Order Modeling of Controlled Flexible Multibody Systems

Thèse de doctorat
présentée en vue
de l'obtention du grade de
Docteur en Sciences Appliquées

par

Olivier Brüls
Ingénieur Civil Electro-Mécanicien (Mécatronique - Productique)
Aspirant F.N.R.S.

Février 2005

Abstract

A mechatronic system is an assembly of technological components, such as a mechanism, sensors, actuators, and a control unit. Recently, a number of researchers and industrial manufacturers have highlighted the potential advantages of lightweight parallel mechanisms with respect to the accuracy, dynamic performances, construction cost, and transportability issues. The design of a mechatronic system with such a mechanism requires a multidisciplinary approach, where the mechanical deformations have to be considered. This thesis proposes two original contributions in this framework.

(i) First, a modular and systematic method is developed for the *integrated simulation of mechatronic systems*, which accounts for the strongly coupled dynamics of the mechanical and non-mechanical components. The equations of motion are formulated using the nonlinear Finite Element approach for the mechanism, and the block diagram language for the control system. The time integration algorithm relies on the generalized-α method, known in structural dynamics. Hence, well-defined concepts from mechanics and from system dynamics are combined in a unified formulation, with guaranteed convergence and stability properties. Several applications are treated in the fields of robotics and vehicle dynamics.

(ii) Usual methods in flexible multibody dynamics lead to complex nonlinear models, not really suitable for control design. Therefore, a systematic *nonlinear model reduction technique* is presented, which transforms an initial high-order Finite Element model into a low-order and explicit model. The order reduction is obtained using the original concept of Global Modal Parameterization: the motion of the assembled mechanism is described in terms of rigid and flexible modes, which have a global physical interpretation in the configuration space. The reduction procedure involves the component-mode technique and an approximation strategy in the configuration space. Two examples are presented: a four-bar mechanism, and a parallel kinematic machine-tool.

Finally, both simulation and modeling tools are exploited for the dynamic analysis and the control design of an *experimental lightweight manipulator* with hydraulic actuators. A Finite Element model is first constructed and validated with experimental data. A reduced model is derived, and an active vibration controller is designed on this basis. The simulation of the closed-loop mechatronic system predicts remarkable performances. The model-based controller is also implemented on the test-bed, and the experimental results agree with the simulation results. The performances and the other advantages of the control strategy demonstrate the relevance of our developments in mechatronics.

Acknowledgements

This dissertation is the result of several years of research at the University of Liège, and I feel deeply indebted to a number of people who directly or indirectly inspired its realization.

First, I would like to thank my advisor Professor Jean-Claude Golinval for his precious help and support during those years. My gratitude also goes to Professor Pierre Duysinx for his technical guidance, and for his enthusiastic encouragements. Every discussion with Professor Michel Géradin had a constructive influence on the orientation of this research, I am pleased to acknowledge him. Those three Professors highly contributed to the improvement of the draft manuscript, I am grateful to them for their advice during its elaboration.

I wish to express my thanks to Professor Wayne J. Book for his cheerful welcome at the Georgia Institute of Technology in summer 2004, and for making my stay as profitable as it could be from all points of view. During the stay, I especially appreciated the collaboration with Ryan Krauss, who initiated me with the Ralf test-bed. Our numerous discussions were highly valuable for the elaboration of chapter 6. I also acknowledge the technical support of J.D. Huggins during this experimental work.

The IAP/AMS framework offered great opportunities to cooperate with belgian specialists in mechatronics. The research on the semi-active suspension has been initiated by the teams of KULeuven-PMA and UCL-PRM, which are gratefully acknowledged. In particular, I am indebted to Professor Paul Fisette (PRM) who took the time to provide the data and the complementary results associated with this problem. I am grateful to Denis Joassin, whose master thesis contributed to the modeling of the suspension.

I wish to thank Professor Rodolphe Sepulchre for interesting discussions about the application of control theories to flexible multibody systems.

Professor Philippe Wenger (IRCCyN, France) is also acknowledged for making the Orthoglide data available.

I would like to thank the Oofelie community, in particular the team of Open Engineering, for its invaluable support in the software implementation of the methods presented in this dissertation. Special thanks go to Professor Alberto Cardona and Elisabet Lens (CIMEC-INTEC, Argentina), whose active collaboration made

possible the development of capabilities for mechanism analysis in Oofelie.

The University of Liège has been a stimulating scientific environment for me. For their friendship, I am grateful to my colleagues from the Vibration and Structural Identification group and from the ASMA department.

Many thanks also go to Professors Peter Eberhard, Jean-Claude Samin, Hendrik Van Brussel, and Doctor Gaetan Kerschen, who accepted to participate in the jury of this doctoral thesis.

Finally, I would like to express my deepest gratitude to my family, for its support, and in particular to my parents for their repeated encouragements throughout the preparation of this dissertation. Many friends also supported me during this period, I wish to thank them all individually.

Contents

I

Introduction

Engineers are encouraged by the market demands to the creation of technological products with increasing complexity. Therefore, they generally exploit a modular and hierarchical *system approach*, which suggests that a product is a system, that can be obtained by appropriate combination of several simpler subsystems.

This thesis concerns *mechatronic systems*, whose purpose is to generate a controlled motion. For example, robots, machine-tools or vehicles are mechatronic systems. At a highly abstracted level, they are composed of hardware and software technological components, such as mechanisms, actuators, sensors and control units, as illustrated in Figure 1.1. The interactions between the subsystems are functional: it is assumed that they occur at localized points, namely the ports of the subsystems.

The design problem consists in refining each subsystem description, leading to an optimal system for the task under consideration. Numerous *design methods and tools* are available to assist the engineer in this challenge; they usually rely on *models*, which are simplified and abstracted representations of the reality.

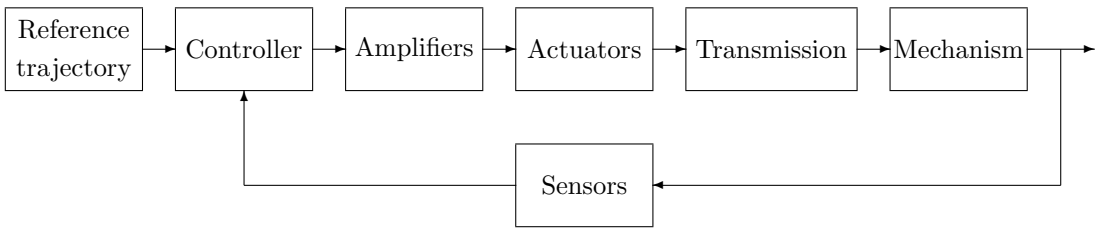


Figure 1.1: A mechatronic system.

Most methods are automated in computer software, able to achieve efficiently and systematically expensive and complex computational treatments.

Traditional design tools are discipline-oriented, and can only be used in a *sequential approach*: each subsystem is designed independently, assuming roughly idealized interactions with the other subsystems. Hence, the initial optimization problem naturally defined at the system level, should be translated at the component level, involving the judgment of the engineer. As pointed out by Van Brussel [VB96], this pragmatic approach obviously leads to a sub-optimal design, and great improvements are expected from a *concurrent approach*, where all the interacting components are considered in a unified framework.

In particular, over the last twenty years, dedicated methods have been developed for the dynamic analysis of *flexible mechanisms*. In a mechatronic context, the mechanism interacts with the *control system*, and the objective of this dissertation is to propose *modeling tools useful for the integrated design of the mechatronic system*. This idea is further developed in the following section.

1.1 Integrated design in mechatronics

The interactions between a mechanism and its control system are highly simplified under the assumptions of (i) a rigid-body behavior, (ii) a simple serial mechanical topology. Therefore, those assumptions are often considered as artificial design constraints, and we shall briefly highlight the potential benefits of their relaxation.

The *rigid-body constraint* implies that the natural frequencies of the mechanism are far beyond the bandwidth of the actuators. Removing this constraint, a lighter mechanical design is possible, with the advantages of a reduced moving mass, a better arm weight to payload weight ratio, a lower power demand, smaller actuators, and a better transportability. A flexible mechanism offers interesting properties to achieve compliant force control. The actuator bandwidth can also be extended, yielding faster motions. However, a subsequent design challenge is to control the flexible dynamics of the mechanism, in order to guarantee tracking performances, stability and disturbance rejection.

Compared to serial topologies, *parallel mechanisms* are appreciated for their higher static and dynamic stiffness. Moreover, the actuators may be located on

the base, reducing the moving mass and simplifying the transmission mechanisms. A difficulty comes from the complexity in the geometrical workspace and in the nonlinear kinematic relation between the motion of the actuated joints and the effector location.

From this early analysis, the advantages of flexible and/or parallel designs appear especially attractive for high-speed applications (*e.g.* high-speed robots and machine-tools), and problems where the mass is a critical issue (*e.g.* large manipulators, space robots and foldable structures). In order to design a controller for those complex dynamic systems, an appropriate *mechanical model* is highly desirable at this level.

For a wide range of practical problems such as car suspensions and traditional machine-tools, the flexibility of the mechanical system is not dominant, and the complex kinematic behavior can be approximately represented by linear equations. After a design based on simplified linear models, the engineer may worry about the validity of the predicted performances. Then a *multidisciplinary simulation tool* may be extremely valuable at a pre-prototyping stage, in order to achieve inexpensive checks and performance predictions before physical implementation and testing.

Even though the rigid body and serial topology constraints are discarded, the sequential design is still a pragmatic approach, which allows a relative independence in the design of the mechanism and of the controller. This sequential approach suffers from inherent limitations which can be overcome by the development of a *concurrent optimization* of the whole system. At this level, a multidisciplinary simulation tool is valuable for the estimation of the optimization criterion and its sensitivities.

Those modeling, simulation, and optimization concepts are relevant at specific stages of the design procedure. In this work, we assume that a first design of the system is available, which includes the description of the mechanism and of the actuators. Hence, the objective is to define the control strategy, and to optimize the remaining design variables. A procedure to solve this problem may involve the following steps:

1. the elaboration of a mechanical model suitable for control design,

2. the design of a control system, relying on the model defined in step 1,

3. the integrated modeling and simulation of the mechatronic system (pre-prototyping),

4. the concurrent optimization of the remaining design variables, based on step 3.

In order to obtain the most appropriate information at each step of the design procedure, various models should be considered. Two important distinctions among modeling concepts are introduced in the next two paragraphs.

Mathematically, the dynamic model of a mechanism is naturally expressed by the *equations of motion*, which define an instantaneous relationship between the external actions (actuator forces, disturbances), the mechanism generalized coordinates and their time-derivatives (velocities and accelerations). The equations of motion, supplemented with a solver algorithm, constitute a *simulation model* of the system, relating time-domain responses (displacements, strains, etc) with time-domain actions (applied forces, commands, etc). In system theory, the same distinction can be done between the state equations, and their formulation with a simulation algorithm.

For complex multibody systems, a modeling software is helpful to formulate automatically the equations of motion from a high-level description. Among the computer modeling methods, *symbolic methods* allow to build the equations of motion in symbolic format, whereas *numerical methods* produce the equations of motion as complex numerical procedures. The symbolic format has the advantages of portability and efficiency, and it provides interesting insights in the analytical structure of the equations. However, numerical methods are able to deal with a more general class of problems, and they are especially suitable to model the dynamics of a flexible mechanism with complex topology in a systematic way.

After this clarification, let us further characterize the modeling requirements in the design procedure, which are directly associated with the *objectives of this research*.

In step 2, the control design usually exploits the mathematical structure of the mechanical equations of motion formulated in step 1. If a linear model is able to capture the essential dynamics, linear control theory can be efficiently applied. Using linearizations at different operating points, it is sometimes possible to generalize this paradigm for nonlinear systems. For the critical applications

considered here, a nonlinear control technique is necessary, which advantageously exploits the structure of a nonlinear model. Since the order of the controller is often proportional to the order of the model, a low-order model in explicit and analytical format is desirable at this stage. However, existing modeling methods for complex flexible mechanisms are not able to construct such a nonlinear model; they lead to high-order and computationally demanding models, which are definitely not appropriate. An important contribution of this thesis is a systematic *reduced-order modeling method*, leading to nonlinear mechanical models, compatible with the requirements of a control design procedure.

Steps 3 and 4 involve a multidisciplinary simulation model, for which less stringent simplicity requirements are imposed: the computational load and the complexity of the model should be balanced with the generality, reliability, modularity and systematic implementation issues. In this thesis, we propose an original *integrated simulation tool for mechatronic systems* based on a unifying Finite Element formulation. This reliable method allows a modular model definition for both the flexible multibody dynamics and the dynamics of the control system.

In the presentation, a specific effort is delivered to define natural connections between the fields of modeling and control. Standard formulations from both areas are combined for the developments of our original mechatronic concepts. Hence, we hope that our point of view can be understood, exploited and developed by specialists from both fields.

1.2 Outline

For the sake of consistency, the dissertation follows a progression from analysis to design concepts, in contradiction with the sequence of the design procedure. Thus, the topics of integrated simulation, model reduction and control design are successively addressed in chapters 4, 5 and 6.

Prior to the presentation of our personal contribution, an extended state of the art is developed in chapter 2. Preliminary modeling and simulation concepts in multibody system dynamics are reviewed before specialized discussions on the simulation of mechatronic systems, the reduced-order modeling and the control of flexible mechanisms. The Finite Element approach proposed by Géradin and Cardona for flexible multibody systems [GC01, Car89, CG89, CG88] is at the core

of our developments, and an overview is presented in chapter 3.

The simulation of mechatronic systems is addressed in chapter 4. The block diagram language is selected for the description of the control system dynamics, and the generalized-α integration scheme is extended to deal with the dynamics of the state variables. After a formal presentation of the simulation method, theoretical convergence and stability results are established. Three examples illustrate the generality and the efficiency of the formulation: a four-bar mechanism, an industrial robot, and a car equipped with a semi-active suspension.

Chapter 5 is devoted to the reduced-order modeling of flexible mechanisms. The method is an extension of the linear component-mode synthesis, which accounts for the nonlinear kinematics of the system. Therefore, the original concept of Global Modal Parameterization is defined, and the reduced equations of motion are formulated. The variations of the model in the configuration space are approximated by piecewise polynomial functions. Critical implementation issues are discussed, in connection with the control design requirements. The method is illustrated with two examples: a rigid parallel machine-tool, and a flexible four-bar mechanism.

In chapter 6, the theoretical tools presented and validated in chapters 4 and 5 are exploited for the modeling and control design of a large and flexible manipulator, which is represented in Figure 1.2. A reduced-order model is validated with experimental results, and the new insights in the dynamics of the manipulator lead to the development of an original control law. Numerical simulations of the closed-loop mechatronic system predict that a compromise can be obtained between the motion bandwidth and the stabilization of the mechanical vibrations, which is confirmed by the experimental results. The various advantages of the control strategy demonstrate the relevance of a global approach in mechatronics.

Our contribution is thus described from the general to the particular: simulation of mechatronic systems, modeling of flexible mechanisms, and application to an experimental manipulator. Finally, chapter 7 draws conclusions and discusses possible directions for future research.

Figure 1.2: Large and flexible manipulator.

II

State of the Art

This research is developed in a multidisciplinary mechatronic framework, and it is thus hopeless to attempt a comprehensive survey of the literature. In order to situate our work, selected topics of interest are reviewed in this chapter: dynamic modeling and simulation of multibody systems, integrated simulation of mechatronic systems, model reduction techniques, and control strategies for flexible mechanisms. On this basis, our strategic choices are motivated for the different contributions of the dissertation.

2.1 Dynamic models for multibody systems

This section discusses existing formulations for the equations of motion in rigid and flexible multibody dynamics, pertaining to the *simulation* and the *control design* problems. Simulation is a direct problem (or forward problem), *i.e.* the equations of motion are solved in the time-domain from given information about the applied forces and driven degrees-of-freedom (dofs). Open-loop and feedforward control are inverse problems, *i.e.* the applied forces and driven dofs are determined in order to yield a specified motion. The design of a closed-loop controller is a non-standard problem, that may exploit the mathematical structure of the equations of motion.

The equations of motion of a mechanism rely on a kinematic parameterization, using a set of generalized coordinates. Sometimes, a unique set of independent coordinates is able to describe arbitrary motion of the system, and the equations of motion are formulated as a set of *Ordinary Differential Equations*

(ODEs). But in general, the kinematic description involves redundant coordinates, interrelated by algebraic and differential constraints. Therefore, the equations of motion are formulated as a set of *Differential and Algebraic Equations* (DAEs), which is less convenient both for control design and simulation.

2.1.1 Rigid mechanisms

A mechanism made of rigid bodies connected by kinematic joints can be described in terms of *relative coordinates*, *i.e.* the angles of the revolute joints and the displacements of the prismatic joints. Usually those coordinates are physically associated with the actuators, which make them relevant for control design, especially for the inverse dynamics problem [AS86]. For tree-topologies, the relative coordinates are independent, which is an additional advantage. Alternatively, *Cartesian coordinates* have been proposed for the efficient and systematic simulation of complex mechanisms [DJB94].

Two approaches can be used in order to obtain the equations of motion: the Newton-Euler formulation, and the Lagrangian formulation.

The *Newton-Euler equations* come from the application of the motion laws to each rigid-body, involving coupling reaction forces and torques. According to d'Alembert's principle, the reaction forces can be eliminated by projection onto the configuration space, leading to a compact set of independent equations. For mechanisms with tree-topology, recursive implementations of the Newton-Euler equations lead to a direct formulation of the independent equations in terms of the relative coordinates, with a high efficiency [LWP80, SF03]. For parallel mechanisms, recursive methods can still be exploited using the cut-joint method to break the loops, and adding closure kinematic constraints between the relative coordinates.

In the *Lagrangian formulation* [DJB94], the system dynamics is described in terms of work and energy using generalized coordinates, *e.g.* relative or Cartesian coordinates. If the generalized coordinates are independent, the reaction forces and torques are automatically eliminated, leading to compact equations of motion. Redundancy in the generalized coordinates requires the introduction of Lagrange multipliers. Optimized recursive Lagrangian formulations have been presented in the literature [Hol80]. Both Newton-Euler and Lagrangian formulations can be

implemented symbolically or numerically.

A detailed review of the literature about the dynamics of rigid mechanisms is presented by Schiehlen [Sch97].

2.1.2 Flexible mechanisms

The kinematic description of a deformed body requires an infinite set of dofs. In order to develop a systematic and general modeling method, an approximation can be obtained by *spatial discretization*. The motion is thereby parameterized by a finite number of coordinates, which are the amplitudes associated with shape functions. Different discretization techniques are possible in flexible multibody dynamics, leading to different formulations of the equations of motion; a few of them are considered here.

The nonlinear Finite Element method

The most general and versatile approach in flexible multibody dynamics is the nonlinear Finite Element method. Each flexible body is divided into elements, and a piecewise approximation strategy is adopted for the displacement field. The amplitudes of the shape functions are nodal parameters, namely translations and rotations with respect to an inertial frame. The consistent description of the large displacements and rotations requires a nonlinear geometric formulation [Sim85, SVQ86]. A systematic methodology for mechanisms analysis using the Finite Element method is proposed by Géradin and Cardona [GC01], who select an updated Lagrangian point of view to handle the large rotations. In order to circumvent the difficult treatment of rotations, Shabana [Sha98] proposed the Absolute Nodal Coordinate Formulation, where the nodal dofs are translations and slopes.

The nodal parameterization of the motion is highly redundant, so that algebraic constraints arise in the equations of motion. The resulting large but sparse system of DAEs is well-suited for simulation, but not for control design. For this reason, alternative approaches leading to simpler models are discussed in the following.

The floating frame of reference approach

According to the floating frame of reference approach [DV76, SW83, MK91, CG91, LHKB91, SM00, Nik03, SF03], the motion of a flexible body is subdivided into a reference motion and a deformation. The former is formally identified with the motion of a rigid floating frame attached to the body, which can be described according to rigid multibody formalisms. The analysis of the deformation within the floating frame is usually simplified assuming small displacements and rotations, so that assumed-mode techniques can be exploited. Three major difficulties are associated with the selection of the reference frame, the definition of the modal shape functions, and the formulation of the coupled kinetic energy.

The floating frame approach was stimulated by simulation problems. However, Book [Boo84] developed a similar method for control design. Relying on a recursive Lagrangian formulation, this method can be implemented numerically or symbolically.

The motion of the reference frame and the deformations are highly coupled by the kinetic energy, requiring a strongly coupled formulation. However, the influence of the deformations on the motion of the reference frame becomes negligible for low-speed applications, so that a weakly coupled scheme can be developed in a co-simulation configuration [VA99, Arn04]: the dynamics of the floating frame is simulated in a software dedicated to rigid multibody systems, whereas the deformations are analyzed in a general purpose Finite Element code.

Lumped approaches

According to a lumped approach, a flexible body is described as a lumped model composed of rigid bodies interconnected by springs and dampers. Several methods have been proposed for simulation by Huston [Hus91] and Wittbrodt and Wojciech [WW95], and for control design by Seto and co-workers, *e.g.* [SH04].

The dynamic analysis can follow any standard methodology for rigid multibody systems. However, the typical number of elements required to model a deformable structure is high, so that the efficiency of the method becomes questionable for a complex three-dimensional mechanism.

2.2 Simulation of multibody systems

The numerical simulation of a distributed system requires a space-time discretization of the partial differential equations.

In the previous section, the nonlinear Finite Element method was recommended for the systematic spatial semi-discretization of a flexible multibody system. The dynamics is then represented by a large but sparse set of index-3 DAEs[1].

The time-discretization and the definition of integration formulae are the next step in the development of a simulation method. Since linear structural dynamics is a special case of flexible multibody dynamics, it is instructive to review the numerical integration methods developed in this field.

2.2.1 Time-integration in linear structural dynamics

A Finite Element model of a dynamic structure is typically a large set of ODEs. These equations accurately represent the low-frequency modes of the physical system, but their high-frequency content is dominated by numerical phenomena resulting from the Finite Element discretization. Due to those numerical modes, the equations of motion are *stiff*[2].

At this level, it is necessary to distinguish explicit and implicit integration schemes, as explained by Géradin and Rixen [GR97]. The stability of an explicit integration scheme is only guaranteed if the time step is small enough with respect to the natural frequencies of the system. On the other side, the commonly used implicit algorithms are *unconditionally stable* (or A-stable), which means that the numerical solution is stable whatever the frequency content of the mechanical system. This is a desirable property for the reliable simulation of stiff equations of motion, but the price to pay for an implicit method is an increased computational complexity.

According to Hughes [Hug87], a time-integration scheme in structural dynamics should ideally combine the following properties: unconditional stability

[1] The index characterizes the algebraic structure of a set of DAEs. It is the minimum number of times that all or part of the equations must be differentiated with respect to time in order to obtain a set of ODEs [BCP96].

[2] The concept of *stiff* differential equations is defined and discussed in details by Hairer and Wanner [HW91].

for linear systems, no more than one set of implicit equations to be solved at each step, second-order accuracy, controllable numerical dissipation in the higher modes, and self-starting procedure.

One-step methods are attractive since they are self-starting and lead to simple code architecture. Among them, the *Newmark family of implicit solvers* have been extensively applied in structural dynamics. From the original formulae proposed by Newmark [New59], improved algorithms have been constructed, such as the Hilber-Hughes-Taylor algorithm [HHT77], and the generalized-α method by Chung and Hulbert [CH93], in order to combine high-frequency dissipation with second-order accuracy. Since unconditional stability can be demonstrated, those methods satisfy the specifications proposed by Hughes.

The Newmark schemes are well adapted for the second-order differential equations of a structural model. Alternatively, general purpose ODE-solvers have been developed by mathematicians, such as the *one-step, multistage Runge-Kutta methods* [Gea71, HNW87]. These solvers can reach a high-order of accuracy, with guaranteed stability properties. Analysts from structural dynamics are sometimes reluctant with respect to this approach, fearing the computational burden of a multi-stage algorithm. However, Owren and Simonsen [OS95] have shown that a class of Runge-Kutta methods could be implemented with an acceptable computational cost. They concluded that those algorithms are also good candidates for the simulation of dynamic structures.

2.2.2 Time-integration in flexible multibody dynamics

An additional difficulty for the simulation of flexible mechanisms comes from the DAE nature of the equations of motion. Brenan *et al.* [BCP96] analyze the conditions under which standard ODEs solvers lead to acceptable solution when applied to DAEs. They present the implementation of multistep backward difference methods in DASSL, a general purpose code designed for index-0 and index-1 DAEs. For higher index DAEs, several implementation difficulties are reported, and the authors recommend a reformulation of the equations.

Following this advice, the index-3 equations of motion should be modified according to a constraint elimination or an index reduction. The *constraint elimination* [WH82, DJB94] is a projection of the initial DAEs on an underlying

set of ODEs, whose solution computed with ODEs solvers naturally satisfies the constraints. The *index reduction* involves the differentiation of the constraint equations, in order to express them at the velocity level (index-2 formulation), or at the acceleration level (index-1 formulation). Unfortunately, the numerical solution of the reduced-index system does not fulfill the original constraints at the displacement level and, due to error propagation, a drift-off from these constraints occurs. To avoid this phenomenon, a stabilization technique [Bau72] should be implemented, which usually brings artificial high-frequencies in the response.

Géradin and Cardona [Car89, CG89, GC01] analyzed the behavior of the Newmark, Hilber-Hughes-Taylor and generalized-α algorithms when directly applied to the original equations of motion. The iterative nature of these implicit algorithms allows to satisfy the algebraic constraints with a level of accuracy specified by the user. In the linear regime, they demonstrated that a weak instability is expected in the Lagrange multipliers, which can be eliminated by adding a small numerical dissipation over the high-frequencies. Those methods have been implemented in the Samcef-Mecano commercial software [SAM99]. Even though no stability proof is available in the nonlinear regime, this methodology turned out to be very efficient and powerful for the wide range of applications encountered by the users of Samcef-Mecano.

Recently, several authors developed a class of algorithms that naturally preserve invariants of the dynamic system, such as energy and canonical momenta [STW92, BBT03, LCG04]. If energy preservation does not necessary leads to the accuracy of the solution, those algorithms usually perform remarkably well. In order to solve efficiently stiff problems, Bauchau *et al.* [BBT03] introduced high-frequency dissipation, leading to a so-called energy-decaying scheme, with guaranteed unconditional stability in the nonlinear regime. The construction of the integration formulae relies on the Lagrangian structure of the mechanical system. In a mechatronic framework, it is usually not possible to describe the control system with a Lagrangian function, and unfortunately, those methods become irrelevant.

As a conclusion, among time-integration methods, *improved Newmark algorithms* (Hilber-Hughes-Taylor and generalized-α methods) turn out to be good candidates for the numerical simulation of mechatronic systems with flexible mechanisms.

2.3 Integrated simulation of mechatronic systems

The functional decomposition of a mechatronic system into simpler interacting subsystems is of practical interest for design purpose. Modularity is also a key issue for the integrated simulation of the mechatronic system. This section addresses the comparison of existing *modular simulation concepts* based on three fundamental questions:

1. how are the subsystems described?
2. how are the couplings between the subsystems defined?
3. how does the simulation algorithm deal with the coupled problem?

Before this comparison, let us analyze general properties of simulation methods.

Basic simulation concepts

In this work, we are especially interested in computational modeling techniques, which have followed the exponential development of computer capabilities over the last decades. Any high-level operation of the user is supported by numerous modeling concepts and dedicated treatments implemented at the lower levels. However, the high-level description should be sufficiently intuitive, so that the underlying levels can be ”forgotten” in the modeling process. For example, modern multibody dynamics software are equipped with powerful graphical user interface, so that the user manipulates visually the idealized bodies and joints of the model, intuitively associated with the actual system.

Hence, a modular simulation software for mechatronic systems requires a *user-friendly interface*, associated with a *consistent low-level implementation*. Most existing software are characterized by the standard architecture sketched in Figure 2.1. This overview is certainly simplified, and a more detailed description would include other libraries, and highlight the deep connections between them. The lower level is the programming language. The compiler, the operating system and the computer hardware could have been mentioned below, but they will not be considered here. At the kernel level, the concepts are built using the syntax of the programming language, and are exploited to establish the element and analysis libraries. The element library aims at describing every subsystem

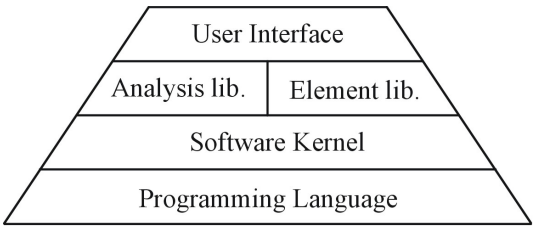


Figure 2.1: General structure of a modular simulation software.

and its contribution to the global equations of motion, whereas the analysis module defines the treatments to solve those equations. At the top, the user interface gives access to the element and analysis libraries in order to construct the simulation model.

In the following, several modular simulation approaches are successively considered: the Finite Element method, physical modeling techniques (the bond graph and the linear graph), variational principles, mathematical approaches (the block diagram). We shall discuss their ability to represent the dynamics of a flexible mechanism within the mechatronic system. A general comparison is summarized in Tables 2.1 and 2.2: the answers to the three fundamental questions (elementary description, coupling strategy and time-integration) are decomposed according to the software architecture given in Figure 2.1. We encourage the reader to refer to them during the presentation of the different methods.

The Finite Element method

The relevance of the Finite Element method for the mechanical modeling of complex flexible multibody systems has been demonstrated earlier. Here the method is presented in a more general framework as a modular modeling approach for distributed systems.

Modularity comes from the geometric decomposition of the distributed system into simple elements coupled by nodal dofs. The *numerical assembly* of elementary contributions leads to the formulation of the coupled equations of motion. Several time-integration algorithms have been proposed in the literature for the simulation of Finite Element models, see section 2.2.

The user interface specifies a high-level syntax for the geometric and physical description of the elements, and for the definition of the nodes. The user is also

able to select the time-integration algorithm and its numerical parameters. At a lower level, each element is defined by its dofs and the algorithms to compute its numerical contribution to the dynamic equilibrium. The integration schemes are implemented at the analysis level, whereas the general concepts of element, dof and node are defined at the kernel level.

Traditionally, the Finite Element method is applied for the simulation of mechanical systems, but it can be naturally extended to represent non-mechanical systems. For example, in a previous work [BG03, BDG03], the dynamics of a controller has been considered in a Finite Element model in order to analyze the active vibration control of a complex mechanical structure. The design of the control system could thus be guided by the simulation results.

Physical modeling: Bond Graph / Linear Graph

In physical modeling, one assumes that each physical component exchanges energy with its environment through localized ports. Hence, the interaction between subsystems is characterized by port variables, which are physical quantities satisfying *conservation laws* at the system level (*e.g.* power, voltage, current, flow, efforts...). For instance, the conservation equations of an electrical network are the Kirchhoff equations, directly connected with the topology of the graph describing the network. Those equations should be complemented by the *constitutive equations* of each component (*e.g.* Ohm's law of a resistor) to characterize the dynamics of the system. By exploiting a graph analysis algorithm, it is possible to organize the equations in a compact form, thereby eliminating redundant variables. This operation can be achieved symbolically, leading to an explicit analytical expression of the minimal set of equations.

Additional information may be given by the user in order to help the graph analysis algorithm. For bond graphs, this information consists in a causality assignment of the interactions, whereas for linear graphs, the user should specify a topological tree. Since the conservation equations are implemented at the kernel level, the development of a new element only requires the definition of its constitutive equations, which is fairly simple.

Physical modeling methods are especially well-suited when the port variables are scalar quantities. Among the numerous references in the literature, we may cite [OE97, KMR00] for bond graphs and [KTKH67] for linear graphs. Recently,

a significant effort has been spent for the standardization of multidisciplinary simulation techniques, leading to the development of the unifying modeling language Modelica [EMO98, Til01], which is based on bond graph theory.

Several authors have extended the concept of bond graph and linear graph for multibody systems, see [Kar97, Fav97] and [McP96, McP98, McP03, SM03a], respectively. According to the comparison study by Sass [Sas04], the linear graph is more appropriate in that case, and it is noticeable that the topological analysis automatically solves the kinematic problem. The extension of the linear graph method for flexible multibody systems was also presented in [SM00, SMH01], relying on an *a priori* spatial discretization with assumed-modes. McPhee *et al.* [SM03a, SM03b] demonstrated the relevance of the linear graph method in electromechanics.

Methods based on variational principles

Several authors [HP95, MEFK97] applied the virtual work principle for the modeling of electromechanical systems. Its scalar formulation is a definite advantage to mix various contributions from electrical and mechanical subsystems. The method relies on the selection of a *minimum set of independent generalized coordinates*, able to describe the electromechanical configuration of the system. For serial rigid mechanisms, this principle leads to the standard Newton-Euler equations in terms of independent joint coordinates. Hadwitch and Pfeiffer [HP95] applied the virtual work principle to rigid multibody systems interacting with electrical networks. From electrical and mechanical generalized coordinates, all other redundant mechanical and electrical variables can be deduced using the kinematic equations and the Kirchhoff laws, respectively.

Mathematical modeling: Block Diagram

In the aforementioned methods, the dynamic equations are formulated from physical principles. However, the input/output behavior of a controller follows an algorithm designed in an artificial way. This is probably the reason why control engineers prefer the more general block diagram formalism, where the equations of the subsystems are directly manipulated. The block diagram is a mathematical language for the modular description of dynamic systems in terms of input,

output, and internal variables; it should not be associated with a particular simulation method. Depending on the coupling strategy, three different approaches are considered below.

The *weakly coupled strategy*, available in several commercial software (Simulink, Vissim, etc.) is often regarded as the standard approach for the simulation of a block diagram model. Each individual subsystem is usually represented by ODEs, but algebraic equations come from input/output interconnections. The time-integration is conducted separately for each block in an asynchronous way, and the coupling results from the exchange of input/output numerical values. The sequence followed for the successive treatments of the blocks is obtained from a causality analysis, itself relying on the input/output causality property of every block. However, this causality analysis breaks down in case of algebraic loops[3], so that reliable algebraic loop detection and solver algorithms are essential for consistent simulation results. As represented in Table 2.2, the ODE solver directly deals with the individual blocks, so that the separation between the element and analysis library is virtual. For this reason, the weak coupling strategy is not suitable for globally implicit integration schemes, and it is not recommended for stiff problems.

Partitioned simulation methods can also be formulated mathematically in terms of block diagrams. Each subsystem can hide a highly complex dynamics (*e.g.* the mechanical or the electrical part of a mechatronic system), and a partitioned treatment intends to improve the computational efficiency and the modularity of the simulation. Among those methods, let us mention the multi-rate integration strategies for ODEs suggested by Gear and Wells [GW84], their generalization by Schiehlen and co-workers to DAEs with algebraic loops [RS98, KS00], and the methods based on Gauss-Seidel iteration for systems coupled by algebraic constraints developed by Arnold *et al.* [Arn01, HAV03]. *Co-simulation* techniques are also partitioned methods, where the subsystems are implemented in different specialized software [VBSV99, VGVDS$^+$99]. In this case, a master linker module manages the task scheduling, and the calling procedures of the other software.

Finally, the block diagram language leads to a *strongly coupled simulation* if

[3]An algebraic loop is a topological loop where all the blocks are direct-feedthrough, which means that their outputs are instantaneously affected by their inputs.

the coupling is imposed prior to the integration scheme: an assembly procedure is necessary to build the coupled equations, which are treated by a monolithic solver. Sass [Sas04] recently adopted this point of view to establish symbolically the coupled dynamic equations of an electromechanical system composed of two subsystems: a rigid mechanism and an electrical network. The strongly coupled approach is also selected in this thesis, but the assembly is performed numerically, as in the Finite Element method.

A few methodologies escape to the strict classification between weak and strong coupling, such as the fastest-first approach evoked in [GW84], or the subcycling technique proposed by Cardona [Car89] for the dynamic analysis of mechanisms with hydraulic actuators. In both cases, smaller time-steps are applied for the fast subsystem according to a weakly coupled approach, but a strong coupling is considered at every global time-step.

Approach selected in this work (Table 2.3)

Chapter 4 will be devoted to the simulation of mechatronic systems composed of a complex flexible mechanism and a control system.

The nonlinear Finite Element method is appropriate to deal with the mechanical problem, whereas the block diagram approach seems especially suitable for the modeling of the control system. Thus, we propose to integrate the block diagram language into the Finite Element framework, so that the *strongly coupled* equations are obtained using the standard Finite Element assembly procedure.

This choice offers the following advantages:

- the detailed, accurate and reliable dynamic representation of the flexible mechanism,

- the generality of the block diagram formulation, which can be extended to discontinuous and sampled systems,

- the modularity of the block diagram library, leading to an intuitive modeling process,

- the formal equivalence with widespread block diagram software, such as Simulink, so that the modeling concepts are very natural for the control engineer, and the exchange of models is facilitated,

		Element description	**Coupling strategy**	**Time-integration**
Finite Element method	**User interface**	Element type, geometry, physical properties	Nodes	Solver selection
	Element library	Numerical contributions	Element Dofs	
	Analysis library			Integration scheme
	Kernel	Element concept	Dof + node concepts, assembly procedure	
Physical modeling	**User interface**	Element type, physical properties	Ports, causality assignment/tree selection	Solver selection
	Element library	Constitutive equations	Port variables	
	Analysis library			ODE/DAE solver
	Kernel	Element concept	Port concept, topological equations, graph analysis	
Variational principles	**User interface**	Element type, physical properties	Generalized coordinates definition	Solver selection
	Element library	Energetic contributions	Generalized/redundant coordinates	
	Analysis library			ODE/DAE solver
	Kernel	Element concept	Relation generalized/ redundant coordinates	

Table 2.1: Comparison between modular simulation concepts.

		Block description	Coupling strategy	Time-integration
Weak coupling	**User interface**	Mathematical model	I/O connections	Solver selection
	Element library	ODEs	I/O variables	Feedthrough nature
	Analysis library			ODE solver
	Kernel	Block concept	I/O variable concept	Block order selection, algebraic loop detection and solver
Partitioned methods	**User interface**	Subsystem model	I/O connections	Solvers/scheduler selection
	Element library	Dedicated methods	I/O variables	ODE/DAE solver, extrapolation method
	Analysis library			Scheduler
	Kernel	Block concept	I/O variable concept	
Strong coupling	**User interface**	Subsystem model	I/O connections	Solver selection
	Element library	Dedicated methods	I/O variables	
	Analysis library			DAE solver
	Kernel	Block concept	I/O variable concept, assembly procedure	

Table 2.2: Comparison between modular simulation concepts: block diagram approaches. "I/O" means "Input/Output".

	Element		Coupling	Simulation
	Mechanism	System		
User interface	Type, geometry, physical properties	I/O dofs, mathematical model	Nodes, I/O dofs	Solver selection
Element library	Numerical contributions	ODEs	Element Dofs	
Analysis library				Coupled DAE solver
Kernel	Element concept	Block diagram concept	Dof/node concepts, numerical assembly	

Table 2.3: Approach selected in this work.

- the reliability of the method, since the strongly coupled approach prevents any difficulty due to algebraic loops.

Two important contributions of this thesis are the formulation of the block diagram concepts in a Finite Element framework, and the extension of the generalized-α method for the time-integration of the coupled equations.

2.4 Model reduction techniques for control design

Modern formalisms in multibody dynamics allow a detailed and reliable representation of complex mechanical systems. However, high levels of accuracy and generality can only be reached at the price of more sophisticated models, which require increased computational resources. Therefore, Eberhard and Schiehlen [ES98] advocated a hierarchical modeling approach, which suggests that different models should be developed to meet the specific requirements at the different stages of a design procedure.

This work addresses the control design of flexible mechanisms. At this design stage, it is commonly accepted that an ideal mechanical model should combine

the following properties:

1. accurate input/output representation in the domain of interest, which is defined from the actuator bandwidth viewpoint and from the workspace viewpoint,

2. low-order, since the order of the controller is often related with the order of the model,

3. represented by ODEs (independent coordinates, no constraint equations),

4. available in closed-form - at least in a structured format, since the structure of the model opens the range of control strategies that can be considered,

5. computationally efficient, since it might be implemented in real-time, or involved in an optimization procedure,

6. obtained using a general and systematic formulation, which is especially valuable for complex systems,

7. portable, in order to be available in software environments dedicated to control design.

The objective of this section is to compare existing modeling techniques with respect to those criteria.

For *rigid mechanisms* with serial topology, an appropriate inverse dynamic model can be obtained using recursive methods. Due to the presence of algebraic constraints, the efficient inverse dynamics of rigid mechanisms with complex parallel topology is still an open problem [BDG06].

In flexible multibody dynamics, the nonlinear Finite Element method and the floating frame approach were pointed as relevant modeling techniques in section 2.1.2. For complex mechanisms with *linear elastic deformations* and *parallel topologies*, those approaches are compared in Table 2.4, according to the seven aforementioned criteria.

Both methods lead to a sufficiently accurate model. However, the Finite Element parameterization involves a large number of redundant coordinates, leading to a high-order and computationally demanding model. As a numerical method, it suffers from a poor portability, and its strongest advantage in this context follows

	Finite Element	Assumed-mode
Accuracy	yes	yes
Order	no	$\sim$
ODEs	no	$\sim$
Structure	$\sim$	yes
Computational efficiency	no	$\sim$
Generality	yes	yes
Systematic formulation	yes	no
Portability	no	yes

Table 2.4: The assumed-mode technique vs. the Finite Element method for flexible mechanisms with complex topologies and linear elasticity.

from its systematic formulation. The floating frame of reference approach relies on a more compact parameterization, with a separation between rigid and flexible coordinates. The resulting model is more structured, which is an advantage for the design of the control algorithm. Symbolic implementations of assumed-mode methods lead to high portability and efficient computations. But in case of complex parallel mechanisms, due to the large number of modes (several modes are necessary for every flexible body), and due to the presence of nonlinear algebraic constraints, the resulting model is still too complicated for control design. Moreover, the definition of the mode shapes cannot be realized in a truly systematic way.

A possible breakthrough may come from reduction or approximation techniques, whose developments in linear and nonlinear frameworks are discussed in the following.

2.4.1 Reduced-order modeling of a linear system

A linear reduction technique transforms an original model into a lower-order model, with a minimized loss in accuracy. The control system is expected to be robust with respect to the omitted dynamics. Several methods were proposed in both fields of linear system theory and linear structural dynamics. We shall review them before considering the reduction of a nonlinear flexible multibody

system.

Several *state-space reduction methods* have been proposed for linear time-invariant systems. The reduction relies on two low-dimensional subspaces: the former defines a coordinate transformation, and the second, a projection operator for the state equations. The objective of a reduction technique is to select those subspaces in order to minimize the accuracy loss. Some methods, well-suited for large-scale problems, are based on Krylov subspaces [GVVD04], and others rely on a truncated balanced realization, as presented by Gawronski [Gaw98]. In order to preserve physical properties of the system, such as dissipativity, congruence transformations[4] may be advantageous, especially for passive RLC electrical circuits and passive mechanical systems [PLS03].

In *linear structural dynamics*, dedicated reduction techniques exploit the second-order and lightly damped nature of the equations of motion. Initially, those methods were not developed for control applications, but for substructured analysis of complex mechanical systems: the reduced contributions of each substructure are established independently in a first step, in order to simplify the analysis of the assembled structure in a second step. A congruence transformation is defined in the space of generalized coordinates as a mode shape matrix. Hurty [Hur65], Craig and Bampton [CB68], Herting [Her79] and Craig [Cra87] proposed various methods to select the truncated modal basis. They are usually denoted *component-mode techniques*, and a more detailed presentation will be given in chapter 5. Interesting combinations of reduction methods in system theory and in structural dynamics have been investigated by Gawronski [Gaw98] and De Fonseca [DF00].

The component-mode synthesis leads to an optimal modal representation of a flexible body, which can be exploited in a floating frame formulation, leading to the concept of superelement [SW83, CG91, GC01]. But again, for a complex flexible multibody system, the resulting model does not meet the requirements of a control design procedure.

[4]In a congruence transformation, the subspaces for the coordinate transformation and for the projection are identical.

2.4.2 Approximated representation of a nonlinear model

A linear dynamic system can be represented using a few matrices (*e.g.* state-space matrices or mass, damping and stiffness matrices), whereas a nonlinear dynamic system may have a very complex structure, which can hardly be casted in a generic format. Sometimes, a nonlinear system is represented explicitly by analytical ODEs, which constitute an appropriate basis for control design. However, the model of a complex flexible mechanism is formulated implicitly in terms of a costly and non-portable numerical procedure. Therefore, it is highly relevant to seek for an approximated explicit formulation, *i.e.* to build a simplified meta-model of the initial model. In the field of optimization, this concept has been applied for response surface techniques [MM95], in order to reduce the number of runs of a full model. In robotics, a nonlinear dynamic model can be pre-computed offline and stored in a look-up table, as discussed by Raibert [Rai77]; this look-up table can then be implemented online for computed-torque control. More general approximation methods have been developed in the field of control, and a quick overview is presented in the following.

Artificial neural networks are theoretically able to represent any kind of non-linearity, and they offer an interesting solution for the approximation of dynamic systems. During a training stage, the neural network learns an input/output behavior from a set of data collected on the initial model. The difficulties of this general method are associated with the choice of the structure of the network, the implementation of the training algorithm, and the black-box nature of the resulting model. The development of neural networks for the control of mechanisms has been considered by Gutiérres *et al.* [GLL98].

Besides neural networks, other simpler approximation functions are possible, such as low-order polynomials, rational functions or kriging functions. In every cases, the data necessary to build the model should be selected with great care, and theories have been specifically developed for this *design of experiment* problem [Mon97]. In particular, adaptive techniques, where the set of data is constructed iteratively in order to minimize the approximation error, have a clear superiority over regular gridding techniques.

In many practical situations, a dynamic system can be characterized by its linearized behavior around specific operating points. For instance, around a con-

figuration of a mechanism, the linearized equations of motion contain relevant information about its dynamic behavior. Hence, a global model might be obtained by appropriate combination of local models, according to the *local model network approach*, also denoted the multiple model approach [TS85, Nel99, MS94, Ang01, Sch01]. Takagi and Sugeno [TS85] proposed to build a smooth global model using concepts from fuzzy sets theory, whereas other authors adopted interpolation methods. Actually, a local model network can be interpreted as a Linear Parameter Varying system, which belongs to the more general class of Linear Time-Varying systems. The parametric dependence can be expressed in several ways, leading for instance to Polytopic Linear Models analyzed by Angelis [Ang01], or models based on Linear Fractional Transformation considered by Scherer [Sch01]. Control theories have been proposed exploiting these specific structures; Caswara and Unbehauen [CU02] and Symens [Sym04] applied some of them for the control of flexible manipulators. However, the combination of local models established for different operating points may lead to a non-consistent global model, especially if the dynamic coordinates (*e.g.* the state variables) do not have a global physical interpretation.

2.4.3 Nonlinear reduction in flexible multibody dynamics

The model associated with an assembled flexible mechanism is nonlinear, but around a given configuration, it can be locally characterized by a linear model, complemented by a quadratic gyroscopic term.

An original nonlinear reduction technique is presented in chapter 5, which extracts a low-order and simplified model from an initial Finite Element model. The reduction relies on the assumption of a linear elastic behavior, and it proceeds in two steps:

- The order reduction results from a *Global Modal Parameterization*, which is inspired by the component-mode technique, with the conceptual differences that the mode shapes are associated with the whole mechanism, and depend on the mechanical configuration. The clear physical interpretation of the modal coordinates guarantees the consistency of the reduced model in the configuration space. The reduction procedure should be repeated for every configuration changes, and as such, it can hardly be exploited for control

design.

- The nonlinear variations of the reduced-order model in the configuration space are *approximated* using a piecewise polynomial interpolation. This strategy is based on an adaptive configuration space inspection algorithm, which minimizes the computational resources to satisfy a specification on the approximation error.

This systematic and general method leads to a highly compact dynamic representation of complex flexible mechanisms. The resulting model is appropriate for control design, but it can also be used to build a more structured model, *e.g.* a polytopic linear model or a linear fractional model. As a special case, the method is applicable to parallel rigid mechanisms [BDG06].

2.5 Control of flexible mechanisms

This section explores active techniques to reduce vibration problems of mechanisms. These methods are opposed to *passive damping methods*, which rely either on the use of high damping materials, such as composites, or on the implementation of lumped passive dampers, tuned for specific operating conditions. In many situations, the use of damping materials is excluded, and it may be difficult to add lumped passive dampers to the system. *Active damping* is then an interesting alternative, and the design of the control algorithm may lead to higher efficiency for a broad range of operating conditions.

According to the classical design approach, a *rigid mechanism* is driven by collocated joint actuators and sensors. A general assumption in this dissertation is that the number of actuators equals the number of rigid modes; in other words, the mechanism is assumed to be fully actuated. In this case, classical control strategies, such as the computed torque technique [AS86], can be applied if an inverse dynamic model is available in real-time. For mechanisms with complex topologies, our original model reduction method can be advantageously exploited for this purpose [BDG06].

When the rigidity assumption needs to be reconsidered, collocated pairs of sensors and actuators may be added in order to achieve a *direct vibration control* [Mei90, Pre97, Gaw98]. This method is widely exploited for active damping

of large space structures. Another solution is to avoid extra actuators, and use the existing actuators for the simultaneous control of the large amplitude motion and of the vibrations. This is the *indirect vibration control* strategy, which is especially suitable for the control of robots and machine-tools. This approach is cheaper and simpler from a mechanical point of view. However, if one is interested in controlling the tip-position, the non-collocated configuration of the sensors and the actuators complicates the design of the control law. This research focuses on the indirect vibration control problem, which has received a major attention in the literature.

The control methods reported here rely on a model resulting from spatial discretization with a finite number of coordinates. Since a flexible system has, in principle, an infinite number of dofs, the higher order dynamics is neglected, but its interaction with the control system may cause spillover instability. Special care is necessary to prevent the system from such undesirable phenomena.

Most control techniques rely on the structure of a mathematical model, hence, we successively consider methods based on linear approaches, on quasi-linear approaches, and finally on nonlinear concepts; of course, each method is conditioned by the availability of an appropriate model.

2.5.1 Methods based on Linear Time Invariant dynamics

The early experimental work realized by Cannon and Schmitz [CS84] focuses on the end-point control of a single flexible beam rotating around its origin. The dynamic equations of this mechanical system can be reasonably approximated by a linear model, experimentally identified. The authors report that the transfer function from the actuators to the tip position is *non-minimum phase*, which can be explained by the non-collocated configuration of the joint actuator and the tip position sensor. In the following, we shall insist on the consequences of this property for the design of an open-loop or of a closed-loop control law.

A first control design problem is the *command generation problem*. For rigid-body mechanisms, it is often accepted that bang-bang acceleration profiles (*i.e.* piecewise constant acceleration profiles) lead to optimal trajectories. When dealing with flexible mechanisms, the high-frequency content of such profiles may excite the vibrations of the system. Bayo and Paden [BP87] recommended the

use of smoother Gaussian velocity profiles. Further improvements can be obtained using input shaping methods proposed by Singer and Seering [SS90], which consist in pre-filtering the command signal in order to reduce the detrimental effect of mechanical flexibility on the dynamic response. Designed for linear systems, the efficiency of input shaping has been demonstrated in several cases for nonlinear mechanisms [RV02], especially when combined with a nonlinearity compensation feedback [KJT95]. The command generation problem can also be reformulated as an inverse dynamics problem: find the torque function to be applied so that the effector follows a specified trajectory in the time-domain. Solutions obtained by feedforward pole/zero cancellation are not applicable because of the non-minimum phase dynamics. Actually, the solution is necessarily non-causal, which means that the torque should start before the tip motion, and continue after reaching the final position. Bayo [Bay87] proposed an off-line frequency domain method, later extended to multilink mechanisms [DB94], whereas Kwon and Book [KB94] advocated a time-domain approach.

Input shaping and inverse dynamics are open-loop methods, which suffer from model sensitivity, and poor disturbance rejection. Therefore, they should be combined with a *feedback control strategy*. Usually, one is interested in controlling the effector of the manipulator, which calls for a noncollocated end-point feedback design. Cannon and Schmitz [CS84] have successfully implemented an LQG state-feedback controller, optimizing the tip response. However, they pointed a high performance sensitivity with respect to variations of the dynamics, which is attributed to the non-collocated configuration. Wang and Vidyasagar [WV91b] argued that the regulation of the tip position can be improved using a reflected tip position feedback, since the corresponding transfer function benefits from the minimum-phase property.

The essential difficulties encountered for the control of linear slewing beams also appear in the multi-link case. But more general approaches are necessary to deal with the nonlinear dynamics caused by the configuration changes.

2.5.2 Quasi-linear control methods

Even though the dynamics of a multilink mechanism is nonlinear, the validity of a linearized model about a particular operating point is sometimes acceptable,

especially in case of small deformations and reasonable velocities. A pragmatic approach consists in a linear control design, which is expected to be *insensitive to the nonlinearities*. Oakley and Cannon [OC89] proposed an LQR controller for a two-link mechanism, whose parameters are optimized for the nominal configuration. For some applications, a satisfactory solution can be obtained combining a gross-motion controller based on the rigidity assumption, with a linear vibration controller active close to the final position. When many final configurations are possible, one vibration controller should be implemented for each linearized local model, which may become cumbersome. Alternatively, De Fonseca [DF00] has presented a linear H_∞ robust controller for a 3-axis machine tool, where the dependence of the dynamics with respect to the configuration of the machine is treated as a model uncertainty. Stability is therefore guaranteed in the whole workspace, but the author reports that this strategy leads to a rather conservative design when compared to non-robust linear designs.

A better dynamic representation can be obtained using a linear time-varying model (or even a nonlinear time-varying model), instead of a linear time-invariant model. Assuming slow variations of the model parameters, an *adaptive control* strategy [Lan79] can be developed: the basic idea is to construct a stable adaptation procedure for the controller, based on real-time measurements. The variations of the model are supposed to be unpredictable, and the adaptation is typically slow. Hence, adaptive control is best suited to deal with uncertainty in the model, in the payload or in the external forces. It has been applied for the control of flexible manipulators by several authors [YBS89, GLL98, CU02, CYC02].

However, if the dependence of the linear model with respect to a few parameters is characterized *a priori*, and if the parameters are directly available from measurements, it is more relevant to apply a *gain scheduling* control strategy. For instance, the linear quadratic regulator theory can be extended for varying systems, and this approach has been exploited by Meirovitch and Chen [MC95] for the vibration control of flexible space robots. Scott [Sco95] developed a gain-scheduling controller for reconfigurable structures: (i) a set of mechanical configurations is selected, (ii) for each of them a linear controller is designed from a linearized model, (iii) the global controller follows from a smooth interpolation of the local controllers in the configuration space. Of course, the global stability of this strategy cannot be assessed theoretically. Symens [Sym04] investigated the active

vibration control of a flexible beam with varying length, in particular, he developed a low-authority controller in order to bring local damping near the resonance frequencies. Following the hierarchical HAC/LAC design [Pre97, VBS02], this low-authority vibration controller (LAC) is complemented with a high-authority motion controller (HAC), responsible for the attenuation of wide-band disturbances on the rigid motion. Among various scheduling techniques reviewed in his work, Symens points out the necessary trade-off between theoretical stability guarantees and practical performances, as well as the difficult elaboration of a low-order model able to capture the configuration-dependent dynamics. We believe that the systematic reduced-order modeling technique presented in this report gives a positive answer to this last problem.

2.5.3 Nonlinear control techniques

There is no general method for the control of the nonlinear dynamics of a flexible multilink mechanism. However, some tools have been proposed in the literature in order to assist the control engineer in the design procedure.

Optimal control theory is a general framework for the construction of a control policy. The statement of a typical optimal control problem can be expressed as follows: find a feasible control such that the system starting from the given initial conditions transfers its state to the target, and minimizes a performance index. Pontryagin's Maximum Principle and Bellman's Dynamic Programming are two powerful tools for this class of variational problems. For linear systems, standard solutions are available for quadratic performance indexes (*i.e.* the linear quadratic regulator), and for indexes expressed by the H_∞ norm of the transfer function of the system. For a class of rigid mechanisms[5], the minimum-time motion is obtained with bang-bang commands. For more complex cases, the optimal control problem can be solved numerically involving expensive computations. Some general methodologies have been recently proposed for systems represented by DAEs [SP01, SLP03], and for multibody systems [Str98, BCGF04, BC04]. The availability of a computationally efficient dynamic model is always a strong advantage when applying such numerical methods.

[5]Fully actuated mechanisms, free from singularities, without frictional effects and without disturbances.

There exists a special class of systems, called *differentially flat* systems, for which there is a one-to-one correspondence between trajectories of a set of "flat outputs" and full state and input trajectories [FLOR95, VNM98, FLMR99]. This special invertibility property leads to an efficient trajectory planning tool: a trajectory specified in output space can be lifted to the state and input space through an algebraic mapping. Ollivier and Sedoglavic [OS01] exploited this tool for the trajectory planning of a very flexible rod, and Thummel *et al.* [TOB01] for flexible joint robots. Since this formalism requires the analytical structure of the dynamic equations, its systematic implementation for complex flexible mechanisms seems to be a difficult problem.

Feedback linearization is a geometric method to transform a nonlinear system into a virtual linear system by a state transformation and a feedback transformation [SL91, Isi95]. Then, linear control theory can be applied for the virtual linear system. For a rigid robot, this method is equivalent to the classical computed torque technique. Sometimes, feedback linearization is classified among inverse dynamics techniques, since it realizes a cancellation of the nonlinear part of the dynamics. Feedback linearization typically requires a full state estimation, which is not possible if part of the dynamics is not observable at the outputs. This unobservable dynamics is called the zero dynamics and it is interpreted as the system dynamics when the outputs are forced to zero. By definition, a nonlinear system is minimum-phase if its zero dynamics is asymptotically stable. Several authors, such as Wang and Vidyasagar [WV91a], applied those advanced concepts to the indirect vibration control of mechanisms with flexible links, and they concluded that the full state dynamics cannot be linearized. The choice of the outputs has a decisive influence on the stability of the zero dynamics: the collocated joint angles lead to an oscillatory zero dynamics, whereas the noncollocated tip position leads to an unstable behavior [WV91a, DLS93]. The output redefinition concept can also be generalized from the linear case [MPK00]. Feedback linearization relies on an analytical expression of the dynamic equations, so that this tool may be hardly applied for complex parallel mechanisms. Moreover, some authors [SJK97] described the pitfalls of the feedback linearization technique, in particular its lack of robustness, since it may involve useless and expensive cancellation of stabilizing nonlinearities, using their dangerous destabilizing counterpart.

Singular perturbation theory has also been investigated for the control of flex-

ible multilink manipulators [SB88, MC95, GLL98, MPK00, GS00, SV01, CYC02]. This theory assumes the partitioning of the state variables into slow and fast variables, so that the dynamics can be represented by two reduced subsystems: the slow subsystem obtained by neglecting the fast oscillatory motion, and the fast subsystem obtained by freezing the slow variables. A composite control can then be applied, combining control laws established separately for each subsytems. In the context of flexible mechanisms, the elastic coordinates are naturally selected as fast variables. The slow subsystem corresponds to a corrected rigid system (*e.g.* including the static contribution of the flexible variables), whereas the fast subsystem has a linear dynamics, parametrically depending on the configuration. The slow controller can be designed on the basis of well-established joint tracking schemes for rigid manipulators, and the fast subsystems can be stabilized using standard methods for linear systems [SB88, GLL98, GS00, SV01] or for linear parameter-varying systems [MC95, MPK00, CYC02]. This theory is extremely appealing, but, as pointed out by Book [Boo93], the performances are limited by the necessary time-scale separation between the rigid and the flexible controllers.

The *Lyapunov method* is a powerful tool for the stability analysis of nonlinear systems. However, for closed-loop systems, the construction of the control algorithm does not completely follow from this theory and involves engineer judgement and intuition.

If the transfer function between the input and a chosen output is *passive*[6], Lyapunov-type control schemes can be more systematically developed with guaranteed stability. Usually, this method leads to simple and robust feedback strategies. For a flexible mechanism, the passivity property of collocated transfer functions can be easily assessed, leading to a classical joint controller. Defining a passive output that includes deformation effects is a more difficult problem.

Considering plane mechanisms that are not influenced by gravity, Ge *et al.* [GLZ96] proposed an energy-based feedback which includes deformation effects and leads to a stable closed-loop system in the sense of Lyapunov. An attractive property of this strategy comes from the independence of the control design with respect to the system dynamics, so that no model is necessary. However, a drawback comes from its conservatism (*i.e.* its low performances), and no guideline is available to optimize the parameters of the controller.

[6] An exact definition of *passivity* in system theory is given by Sepulchre *et al.* [SJK97].

Specific nonlinear control design methods have been developed for systems with a cascade structure, *e.g.* the states of one subsystem are the control variables for the following. A step-by-step recursive design procedure can then be exploited, such as backstepping or forwarding [SJK97]. Backstepping has been explicitly applied for flexible joint robots [OL99]. Recursivity is also at the basis of the inertial damping concept developed for the control of macro/micromanipulators. In the macro/micro configuration, a small rigid robot is mounted at the tip of a large and flexible manipulator. Several approaches have been proposed for the control of such system, and an extended review is presented by George [Geo02]. Among them, the inertial damping method consists in controlling the motion of the rigid robot in such a way that the inertia forces transmitted to the supporting flexible manipulator have a damping effect on the vibration modes. This is a true cascade system: the control of the micromanipulator motion allows the active damping of the macromanipulator.

2.5.4 Control strategy selected in this work

After this review, the compromise between stability, performances and robustness appears more than ever critical when designing a control law for a flexible mechanism. In chapter 6, the control of an experimental manipulator will be investigated on the basis of the two-time-scale approach: a slow joint-tracking controller is complemented with a fast vibration controller. Related with HAC/LAC control techniques and with approaches developed from singular perturbation theory, this method has several advantages:

- a simple hardware and software implementation,
- an adjustable compromise between the joint tracking performances and the vibration suppression,
- the absence of inversion or cancellation of the nonminimum phase dynamics, so that the performances are less sensitive to modeling errors.

The fast vibration controller relies on an extension of the inertial damping concept, which exploits the particular structure of the inertia forces in a low-order mechanical model. Hence, the nonlinear model reduction technique developed in

chapter 5 will be extremely valuable for the design and the implementation of the control law.

III

The Finite Element Approach in Multibody Dynamics

According to chapter 2, the Finite Element method is appropriate for the modeling and the simulation of complex parallel mechanisms with significant flexibility. This chapter briefly presents the approach developed by Géradin and Cardona [CG88, Car89, CG89, GC01], which is at the center of our developments. The motion parameterization, the dynamics and the time-integration strategy are successively discussed.

3.1 Motion parameterization

In the literature, several coordinate choices have been proposed for the kinematic description of a mechanism, such as:

- the minimal coordinates,
- the relative coordinates, also denoted Lagrangian coordinates or joint coordinates,
- the Cartesian coordinates, or reference point coordinates,
- the Finite Element coordinates, also denoted natural coordinates, or absolute coordinates,
- the mixed coordinates.

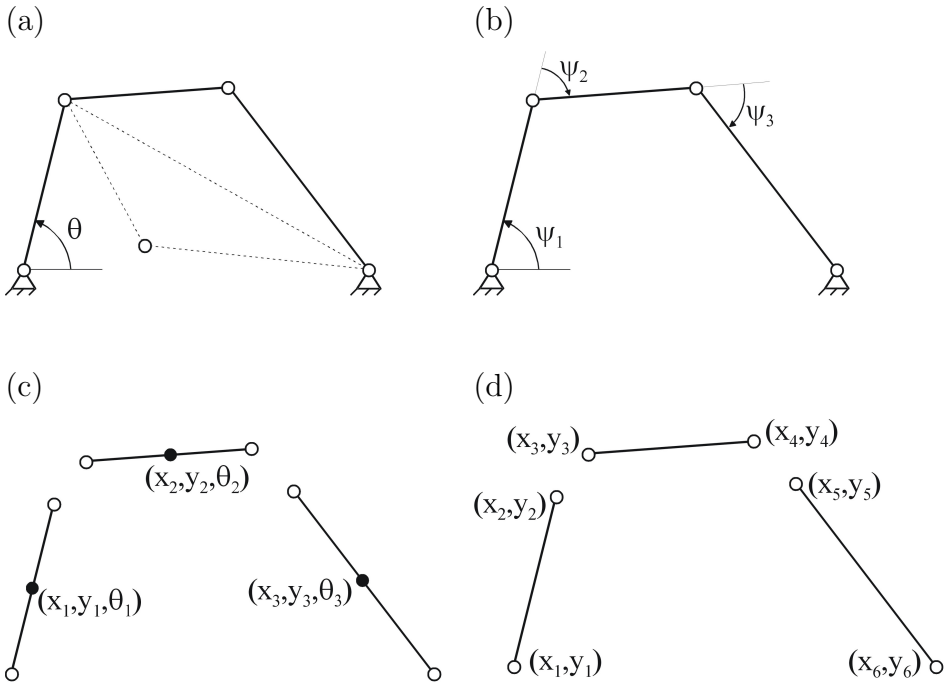


Figure 3.1: (a) Minimal coordinates, (b) Relative coordinates, (c) Cartesian coordinates, (d) Finite Element coordinates or natural coordinates.

We propose to illustrate these concepts for a simple planar mechanism; afterwards, the representation of spatial rotations will be specifically addressed.

3.1.1 Kinematic description of a planar mechanism

The rigid four-bar mechanism represented in Figure 3.1 has one kinematic dof, and its description in terms of minimal coordinates requires a single parameter: the angle θ. However, in Figure 3.1(a), two configurations can be associated with a single value of θ, so that this parameterization is only valid for a restricted part of the configuration space. The strong advantage of the minimal coordinates comes from their independence, *i.e.* the absence of algebraic constraint.

The relative coordinates (Figure 3.1(b)) allow a recursive description of the mechanical configuration. They are especially suitable for mechanisms with a tree-topology, since their independence is then guaranteed. For a parallel mechanism, such as the four-bar example, the relative coordinates are dependent, and

connected by loop-closure constraints.

The Cartesian coordinates (Figure 3.1(c)) are defined as the position and orientation of the center of mass of each body. The connections between bodies are defined *a posteriori*, using appropriate kinematic constraints. This allows a simple and systematic formulation, which leads to a rather large but sparse system of equations.

Those three types of coordinates are competing solutions for the kinematic description of rigid multibody systems. However, in flexible multibody dynamics, the deformations should be described with additional (modal) coordinates, leading to a hybrid set of coordinates. The coupling between the flexible coordinates and the other coordinates generally leads to a difficult formulation of the kinetic energy. In this context, *the Finite Element coordinates* are more appropriate: each body is characterized by a set of nodes, and each node is characterized by its own set of coordinates, see Figure 3.1(d). Let us define l_1, l_2, l_3 and l_4 as the respective lengths of the four bars between the nodes 1-2, 3-4, 5-6 and 1-6. The Finite Element coordinates are redundant, and three types of constraints are to be considered:

- boundary nodal constraints:

$$x_1 = y_1 = y_6 = 0 \qquad \text{and} \qquad x_6 = l_4 \tag{3.1}$$

- assembly nodal constraints, imposed by boolean identification:

$$x_2 = x_3, \qquad y_2 = y_3, \qquad x_4 = x_5, \qquad y_4 = y_5 \tag{3.2}$$

- rigidity constraints, which may be expressed as a constant length requirement:

$$l_i^2 = (x_{2i} - x_{2i-1})^2 + (y_{2i} - y_{2i-1})^2 \qquad i = 1, 2, 3 \tag{3.3}$$

A mixed formulation may involve several types of coordinates. For instance, despite our choice of absolute coordinates, it may be interesting to consider the relative angle θ as an additional dof. Since the parameterization with absolute coordinates is complete, θ is redundant, and connected to the absolute variables by an algebraic constraint:

$$y_2 \ \cos\theta - x_2 \ \sin\theta = 0 \tag{3.4}$$

Mixed coordinates will be extremely useful to develop our reduction technique, which will be discussed later.

3.1.2 Description of spatial rotations

The definition of the Finite Element coordinates is straightforward for a simple planar mechanism, as seen in the previous section, but for a spatial mechanism, the parameterization of the absolute nodal rotation is a challenging problem. Generally, a finite rotation can be represented using a proper unitary or orthonormal 3×3 rotation matrix. Such a matrix can be described by three independent parameters, and several choices have been proposed in the literature, such as the Euler angles, the Bryant angles, the Cartesian rotation vector, the Rodrigues parameters, the conformal rotation vector or the linear parameters. However, all those parameterizations suffer from singularities. The four Euler parameters should also be mentioned, which lead to a non-singular but redundant parameterization.

Due to the various advantages and drawbacks of each parameterization, the optimal parameterization is problem-dependent. Following the conclusion of Cardona [Car89, page 65], the most relevant choices for a Finite Element formulation in flexible multibody dynamics are the Cartesian rotation vector and the conformal rotation vector. The former was at the basis of the Samcef-Mecano software [SAM99]; both of them are combined for the current implementation in the Oofelie software [CKG94], for reasons that are beyond the scope of this dissertation.

3.2 Dynamics

The Finite Element coordinates are absolute coordinates, and the total motion (rigid-body motion and elastic deformation) is directly referred to an inertial frame. Due to the large displacements and rotations of the mechanical elements with respect to this frame, the linear theory of elasticity is not applicable, and a three-dimensional nonlinear theory is necessary.

An updated Lagrangian point of view for the rotation parameters has been recommended by Géradin and Cardona [GC01]. This means that the rotations are described as increments with respect to a previous configuration, so that the

singularities of the rotation parameterization are avoided, full symmetry of the operators is obtained for the conservative loading case, and the time-integrator is quite naturally extended to treat the rotational dofs.

The kinetic and potential energies $\mathcal{K}$ and $\mathcal{V}$ may be formulated using the $n \times 1$ vector of Finite Element coordinates $\mathbf{q}$, and the equations of motion can be derived from the Hamilton's principle:

$$\delta \int_{t_1}^{t_2} (\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) + \mathcal{W})\, dt = 0 \qquad \text{subject to} \qquad \mathbf{\Phi}(\mathbf{q}) = \mathbf{0} \tag{3.5}$$

where the Lagrangian $\mathcal{L}$ is defined by

$$\mathcal{L} = \mathcal{K} - \mathcal{V}, \tag{3.6}$$

$\mathcal{W}$ denotes the virtual work of external non-conservative loads $\mathbf{Q}$:

$$\delta\mathcal{W} = \delta\mathbf{q}^T\, \mathbf{Q} \tag{3.7}$$

and $\mathbf{\Phi}$ are m scleronomic constraints. The formulation may also include non-holonomic constraints, but we shall not insist on this point here.

The problem (3.5) can be replaced by the equivalent formulation:

$$\delta \int_{t_1}^{t_2} (\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) + \mathcal{W} - \boldsymbol{\lambda}^T\, \mathbf{\Phi})\, dt = 0 \tag{3.8}$$

where $\boldsymbol{\lambda}$ is the $m \times 1$ vector of Lagrange multipliers associated with the constraints. Performing the variation and the integration by part, one obtains the Lagrange equations:

$$\begin{cases} \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial \mathcal{L}}{\partial \mathbf{q}} + \mathbf{\Phi}_{\mathbf{q}}^T\, \boldsymbol{\lambda} = \mathbf{Q} \\ \mathbf{\Phi}(\mathbf{q}) = \mathbf{0} \end{cases} \tag{3.9}$$

where $\mathbf{\Phi}_{\mathbf{q}}$ denotes the constraint gradient. Assuming that $\mathcal{K}$ can be expressed as a quadratic form of generalized velocities with a symmetric mass matrix $\mathbf{M}(\mathbf{q})$:

$$\mathcal{K} = \frac{1}{2}\, \dot{\mathbf{q}}^T\, \mathbf{M}(\mathbf{q})\, \dot{\mathbf{q}} \tag{3.10}$$

the equations of motion can be put in matrix form:

$$\begin{cases} \mathbf{M}\, \ddot{\mathbf{q}} + \mathbf{\Phi}_{\mathbf{q}}^T\, \boldsymbol{\lambda} = \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}) \\ \mathbf{\Phi}(\mathbf{q}) = \mathbf{0} \end{cases} \tag{3.11}$$

where the vector of apparent forces $\mathbf{g}(\mathbf{q}, \dot{\mathbf{q}})$ collects external forces, internal forces and complementary inertia forces:

$$g_k = Q_k - \frac{\partial \mathcal{V}}{\partial q_k} - \sum_i \sum_j \left(\frac{\partial M_{ki}}{\partial q_j} - \frac{1}{2} \frac{\partial M_{ij}}{\partial q_k} \right) \dot{q}_i \, \dot{q}_j \tag{3.12}$$

This system of equations, and its linearized counterpart, can be directly used for numerical analysis, *e.g.* in the time domain. Several enhanced formulations have been proposed for more efficient numerical treatments. Among them, we shall review the constraint elimination technique and the augmented Lagrangian method.

3.2.1 The constraint elimination method

The constraint elimination method aims at replacing the initial problem by an equivalent unconstrained problem. It relies on a partitioning of the n generalized coordinates $\mathbf{q}$ into two subsets: the $n-m$ independent coordinates $\boldsymbol{\theta}$, and the m dependent coordinates $\mathbf{q}^*$. Then, we seek for an explicit elimination formula:

$$\mathbf{q}^* = \boldsymbol{\Phi}^*(\boldsymbol{\theta}) \tag{3.13}$$

Since the m algebraic constraints $\boldsymbol{\Phi}$ are nonlinear and implicit, the expression of $\boldsymbol{\Phi}^*$ cannot be constructed analytically. However, the Jacobian of the transformation can be computed by implicit differentiation:

$$\frac{\partial \mathbf{q}^*}{\partial \boldsymbol{\theta}} = -\boldsymbol{\Phi}_{\mathbf{q}^*}^{-1} \, \boldsymbol{\Phi}_{\boldsymbol{\theta}} \qquad \Rightarrow \qquad \frac{\partial \mathbf{q}}{\partial \boldsymbol{\theta}} = \mathbf{q}_{\boldsymbol{\theta}} = \begin{bmatrix} \mathbf{I} \\ -\boldsymbol{\Phi}_{\mathbf{q}^*}^{-1} \, \boldsymbol{\Phi}_{\boldsymbol{\theta}} \end{bmatrix} \tag{3.14}$$

Numerical treatments usually rely on Newton-Raphson iterations of the linearized dynamic equations. At this level, the Jacobian $\mathbf{q}_{\boldsymbol{\theta}}$ can be exploited to transform the initial linearized problem in terms of the redundant variables $\mathbf{q}$ into a reduced linearized problem in terms of the independent variables $\boldsymbol{\theta}$. This technique is explained with more details by De Jalón and Bayo [DJB94].

3.2.2 The augmented Lagrangian method

The objective of the augmented Lagrangian method is to improve the numerical conditioning of the Lagrange multiplier method by adding a penalty term of

moderate amplitude. At the same time, appropriate constraint scaling is introduced in order to generate system matrices of the same order of magnitudes. The augmented functional of Hamilton's principle (3.8) is

$$\delta \int_{t_1}^{t_2} (\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) + \mathcal{W} - k\, \boldsymbol{\lambda}^T\, \boldsymbol{\Phi} - p\, \boldsymbol{\Phi}^T\, \boldsymbol{\Phi})\, dt = 0 \tag{3.15}$$

where k and p are the scaling and penalty factors. The dynamic equations follow:

$$\begin{cases} \mathbf{M}\, \ddot{\mathbf{q}} + \boldsymbol{\Phi}_{\mathbf{q}}^T\, (p\, \boldsymbol{\Phi} + k\, \boldsymbol{\lambda}) = \mathbf{g} \\ k\, \boldsymbol{\Phi}(\mathbf{q}) = \mathbf{0} \end{cases} \tag{3.16}$$

Since the penalty term vanishes at the solution point, it is easily observed that the method provides the exact solution of the initial problem.

Since the time-integration procedure involves a Newton-Raphson procedure, it is important to formulate the linearized equations for the corrections of the displacements, velocities and accelerations:

$$\begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\ddot{\mathbf{q}} \\ \Delta\ddot{\boldsymbol{\lambda}} \end{bmatrix} + \begin{bmatrix} \mathbf{C}_t & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\dot{\mathbf{q}} \\ \Delta\dot{\boldsymbol{\lambda}} \end{bmatrix} + \begin{bmatrix} \mathbf{K}_t & k\, \boldsymbol{\Phi}_{\mathbf{q}}^T \\ k\, \boldsymbol{\Phi}_{\mathbf{q}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{q} \\ \Delta\boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} -\mathbf{res}^{\mathbf{q}} \\ -\mathbf{res}^{\boldsymbol{\Phi}} \end{bmatrix} + \mathcal{O}(\Delta^2) \tag{3.17}$$

where

- $\mathbf{res}^{\mathbf{q}}$ and $\mathbf{res}^{\boldsymbol{\Phi}}$ denote the residual vectors of equilibrium and constraints, with:

$$\mathbf{res}^{\mathbf{q}} = -\mathbf{g} + \mathbf{M}\, \ddot{\mathbf{q}} + \boldsymbol{\Phi}_{\mathbf{q}}^T\, (p\, \boldsymbol{\Phi} + k\, \boldsymbol{\lambda}) \tag{3.18}$$

- $\mathbf{C}_t$ is a tangent damping matrix resulting from the variation of the gyroscopic forces with respect to the velocities:

$$\mathbf{C}_t = -\frac{\partial \mathbf{g}}{\partial \dot{\mathbf{q}}} \tag{3.19}$$

- $\mathbf{K}_t$ is a tangent stiffness matrix resulting from variations with respect to the displacements:

$$\mathbf{K}_t = -\frac{\partial \mathbf{g}}{\partial \mathbf{q}} + \frac{\partial (\mathbf{M}\, \ddot{\mathbf{q}})}{\partial \mathbf{q}} + \frac{\partial \left(\boldsymbol{\Phi}_{\mathbf{q}}^T\, (p\, \boldsymbol{\Phi} + k\, \boldsymbol{\lambda}) \right)}{\partial \mathbf{q}} \tag{3.20}$$

Usually, a good approximation is given by:

$$\mathbf{K}_t \simeq -\frac{\partial \mathbf{g}}{\partial \mathbf{q}} + p\, \boldsymbol{\Phi}_{\mathbf{q}}^T\, \boldsymbol{\Phi}_{\mathbf{q}} \tag{3.21}$$

The penalty term adds a positive definite contribution to the tangent stiffness, which improves the numerical conditioning and the convergence of the Newton-Raphson iterations.

3.3 Time-integration

In chapter 2, the relevance of the Newmark family of implicit algorithms for the simulation of flexible mechanisms was highlighted. This section presents the original Newmark scheme [New59] and its generalized-α extension by Chung and Hulbert [CH93]. Theoretical results will also be detailed, related with the issues of stability and convergence.

3.3.1 Newmark algorithm

The original Newmark integration formula are obtained from a Taylor series expansion of the displacements and velocities, with respect to the time-step size h:

$$\begin{aligned} \mathbf{q}_{n+1} &= \mathbf{q}_n + h\ \dot{\mathbf{q}}_n + h^2\ (\tfrac{1}{2} - \beta)\ \ddot{\mathbf{q}}_n + h^2\ \beta\ \ddot{\mathbf{q}}_{n+1} \\ \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + h\ (1 - \gamma)\ \ddot{\mathbf{q}}_n + h\ \gamma\ \ddot{\mathbf{q}}_{n+1} \end{aligned} \tag{3.22}$$

where the constants β and γ are numerical parameters. It may be shown that the optimal choice of the parameters corresponds to the average constant acceleration formula, obtained with the particular values:

$$\gamma = \frac{1}{2} \qquad \text{and} \qquad \beta = \frac{1}{4} \tag{3.23}$$

They give maximal second-order accuracy, and unconditional linear stability, as will be demonstrated later. Numerical damping can be introduced in the Newmark formula according to:

$$\gamma = \frac{1}{2} + \alpha \qquad \text{and} \qquad \beta = \frac{1}{4}\left(\gamma + \frac{1}{2}\right)^2 \qquad \alpha > 0 \tag{3.24}$$

where α is the numerical damping parameter. This choice allows to increase the numerical damping in the system while remaining on the stability boundary of the integration scheme. A more systematic presentation of the Newmark algorithm in structural dynamics is described by Géradin and Rixen [GR97].

The numerical solution of the dynamic equations (3.16) follows an iterative predictor-corrector scheme, exploiting the linearized equations (3.17). From given

values of $\mathbf{q}_n, \dot{\mathbf{q}}_n, \ddot{\mathbf{q}}_n$ at time t_n, an initial prediction at time t_{n+1} is obtained with the zero acceleration assumption:

$$\begin{aligned} \ddot{\mathbf{q}}^0_{n+1} &= \mathbf{0} \\ \dot{\mathbf{q}}^0_{n+1} &= \dot{\mathbf{q}}_n + h\,(1-\gamma)\,\ddot{\mathbf{q}}_n \\ \mathbf{q}^0_{n+1} &= \mathbf{q}_n + h\,\dot{\mathbf{q}}_n + h^2\,(\tfrac{1}{2}-\beta)\,\ddot{\mathbf{q}}_n \end{aligned} \tag{3.25}$$

and to satisfy the Newmark formulae (3.22), the iterative corrections should satisfy:

$$\begin{aligned} \triangle\ddot{\mathbf{q}}_{n+1} &= \tfrac{1}{\beta\,h^2}\,\triangle\mathbf{q}_{n+1} \\ \triangle\dot{\mathbf{q}}_{n+1} &= \tfrac{\gamma}{\beta\,h}\,\triangle\mathbf{q}_{n+1} \end{aligned} \tag{3.26}$$

so that the linearized system (3.17) becomes:

$$\begin{bmatrix} \mathbf{S}_t^{\mathbf{qq}} & k\,\mathbf{\Phi}_{\mathbf{q}}^T \\ k\,\mathbf{\Phi}_{\mathbf{q}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{q} \\ \Delta\boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} -\mathbf{res}_k^{\mathbf{q}} \\ -\mathbf{res}_k^{\mathbf{\Phi}} \end{bmatrix} \tag{3.27}$$

with the augmented tangent matrix:

$$\mathbf{S}_t^{\mathbf{qq}} = \mathbf{K}_t + \frac{\gamma}{\beta\,h}\,\mathbf{C}_t + \frac{1}{\beta\,h^2}\,\mathbf{M} \tag{3.28}$$

The solution to the system (3.27) is the correction for the Newton-Raphson procedure. The complete integration algorithm is summarized in Figure 3.2.

3.3.2 Generalized-α method

Due to the Finite Element discretization and to the presence of algebraic constraints, the dynamic equations are numerically stiff, and *high-frequency numerical damping* is desirable. However, the stabilized Newmark scheme is only first-order accurate. Several alternatives have been proposed in order to introduce numerical damping at high-frequency *without degrading the order of accuracy*. Among them, the generalized-α method described by Chung and Hulbert [CH93] includes as particular cases the most important integration schemes in structural dynamics, such as the Hilber-Hughes-Taylor algorithm [HHT77], and it is therefore a very general framework for theoretical investigations. It consists of keeping the Newmark formulae (3.22), whereas the residual equations are modified by averaging the different contributions between both time instants:

$$\begin{cases} (1-\alpha_m)\,(\mathbf{M}\,\ddot{\mathbf{q}})_{n+1} + \alpha_m\,(\mathbf{M}\,\ddot{\mathbf{q}})_n + (1-\alpha_f)\,\mathbf{g}^*_{n+1} + \alpha_f\,\mathbf{g}^*_n = \mathbf{0} \\ (1-\alpha_f)\,k\,\mathbf{\Phi}_{n+1} + \alpha_f\,k\,\mathbf{\Phi}_n = \mathbf{0} \end{cases} \tag{3.29}$$

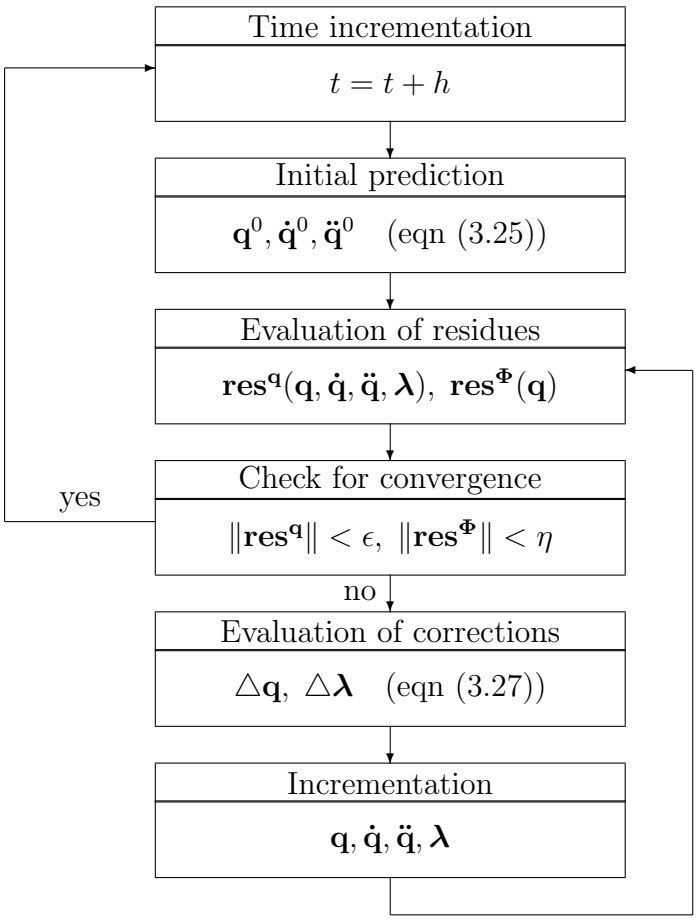


Figure 3.2: Newmark algorithm

where $\mathbf{g}^*$ is a notation for $\mathbf{\Phi}_\mathbf{q}^T\ (k\,\boldsymbol{\lambda} + p\,\mathbf{\Phi}) - \mathbf{g}$, while α_m and α_f are numerical parameters. The Hilber-Hughes-Taylor algorithm is obtained for $\alpha_m = 0$ and $\alpha_f \in [0, 1/3]$. The optimal parameters of the generalized-α method can be computed from the desired spectral radius at infinity $\rho_\infty^\mathbf{q}$:

$$\alpha_m = \frac{2\,\rho_\infty^\mathbf{q} - 1}{\rho_\infty^\mathbf{q} + 1} \qquad \text{and} \qquad \alpha_f = \frac{\rho_\infty^\mathbf{q}}{\rho_\infty^\mathbf{q} + 1} \tag{3.30}$$

Defining $\alpha_{fm} = \alpha_f - \alpha_m$, the Newmark parameters are now given by:

$$\gamma = \frac{1}{2} + \alpha_{fm} \qquad \text{and} \qquad \beta = \frac{1}{4}\left(\gamma + \frac{1}{2}\right)^2 \tag{3.31}$$

It will be demonstrated later that this scheme is second-order accurate even for $\alpha_{fm} > 0$.

The numerical solution is obtained with a predictor-corrector strategy as for the Newmark algorithm. At the correction step, the augmented tangent matrix becomes:

$$\mathbf{S}_t^{\mathbf{qq}} = (1-\alpha_f)\,\mathbf{K}_t + (1-\alpha_f)\,\frac{\gamma}{\beta\,h}\,\mathbf{C}_t + (1-\alpha_m)\,\frac{1}{\beta\,h^2}\,\mathbf{M} \tag{3.32}$$

and the linearized equations:

$$\begin{bmatrix} \mathbf{S}_t^{\mathbf{qq}} & k\,(1-\alpha_f)\,\mathbf{\Phi}_\mathbf{q}^T \\ k\,(1-\alpha_f)\,\mathbf{\Phi}_\mathbf{q} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{q} \\ \Delta\boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} -\mathbf{res}_k^\mathbf{q} \\ -\mathbf{res}_k^\mathbf{\Phi} \end{bmatrix} \tag{3.33}$$

where the residues are computed according to (3.29).

Remark 3.1 *As a consequence of the modification of the residual equation in the generalized-α method, the acceleration $\ddot{\mathbf{q}}_{n+1}$ is a poor first-order approximation for the true acceleration $\ddot{\mathbf{q}}(t_{n+1})$ [EBB02]. Assuming that an order 2 approximation is available for $\mathbf{q}_{n+1}$, $\dot{\mathbf{q}}_{n+1}$ and $\boldsymbol{\lambda}_{n+1}$, an order 2 approximation $\mathbf{a}_{n+1}$ would satisfy the original residual equation at time t_{n+1}:*

$$(\mathbf{M}\,\mathbf{a})_{n+1} + \mathbf{g}^*(\mathbf{q}_{n+1}, \dot{\mathbf{q}}_{n+1}, \boldsymbol{\lambda}_{n+1}) = \mathbf{0} \tag{3.34}$$

Using the first equation of (3.29) to eliminate $\mathbf{g}^*$, an equivalent and useful condition is obtained if the matrix $\mathbf{M}$ is constant and non-singular:

$$(1-\alpha_f)\,\mathbf{a}_{n+1} + \alpha_f\,\mathbf{a}_n = (1-\alpha_m)\,\ddot{\mathbf{q}}_{n+1} + \alpha_m\,\ddot{\mathbf{q}}_n \tag{3.35}$$

3.3.3 Linear stability analysis

When dealing with stiff systems of equations, the stability of the integration scheme is critical for a successful simulation. In flexible multibody dynamics, the presence of algebraic constraints strongly affects the stability of the numerical solution. For unconstrained problems, the stability of the generalized-α method is analyzed by Chung and Hulbert [CH93] in the linear regime, and by Erlicher *et al.* [EBB02] in the nonlinear regime. For constrained problems, linear stability results are presented by Géradin and Cardona [GC01].

In this thesis, all stability analyses are realized for linear systems; as Géradin and Cardona [GC01], we believe that these results are relevant indications of the stability properties for nonlinear systems. This section presents a demonstration of the unconditional stability of the generalized-α method inspired by the work of Géradin and Cardona [GC01]; the unconstrained and constrained cases are successively addressed. In chapter 4, those developments will be extended for the stability analysis of our integrated simulation method.

Unconstrained system

The stability analysis of an integration scheme applied to a linear system of equations:

$$\mathbf{M}\,\ddot{\mathbf{q}} + \mathbf{K}\,\mathbf{q} = \mathbf{0} \tag{3.36}$$

is hereby conducted according to the developments of Chung and Hulbert [CH93]. The problem is simplified by invoking the diagonalization of the equations with a linear transformation. It is straightforward that the numerical behavior will be the same for the initial and the transformed systems. Since each modal equation can be analyzed independently, it is sufficient to consider the stability of the integrator applied to the scalar test equation

$$\ddot{q} + \omega^2\, q = 0 \tag{3.37}$$

According to the generalized-α methodology, the time-discretized expression is

$$(1-\alpha_m)\,\ddot{q}_{n+1} + \alpha_m\,\ddot{q}_n + (1-\alpha_f)\,\omega^2\,q_{n+1} + \alpha_f\,\omega^2\,q_n = 0 \tag{3.38}$$

Complementing this equation with the Newmark formulae (3.22), we get the matrix relation

$$\begin{bmatrix} (1-\alpha_f)\omega^2 & 0 & (1-\alpha_m) \\ 1 & 0 & -\beta h^2 \\ 0 & 1 & -\gamma h \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \\ \ddot{q} \end{bmatrix}_{n+1} + \begin{bmatrix} \alpha_f\omega^2 & 0 & \alpha_m \\ -1 & -h & -(\frac{1}{2}-\beta)h^2 \\ 0 & -1 & -(1-\gamma)h \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \\ \ddot{q} \end{bmatrix}_{n} = \mathbf{0} \tag{3.39}$$

and we seek its eigensolutions, characterized by a synchronous behavior:

$$\begin{bmatrix} q \\ \dot{q} \\ \ddot{q} \end{bmatrix}_n = \varphi_n \begin{bmatrix} q_0 \\ \dot{q}_0 \\ \ddot{q}_0 \end{bmatrix} \qquad \text{with} \qquad \varphi_{n+1} = \zeta\, \varphi_n \tag{3.40}$$

φ is the amplitude, $[q_0 \;\; \dot{q}_0 \;\; \ddot{q}_0]^T$ is the eigenvector, and ζ is the eigenvalue. If we define the polynomials,

$$\begin{aligned} &P_\epsilon(\zeta) = (1-\epsilon)\,\zeta + \epsilon \qquad \epsilon = \{\alpha_m, \alpha_f\} \\ &P_\gamma(\zeta) = \gamma\,\zeta + (1-\gamma) \\ &P_\beta(\zeta) = \beta\,\zeta + (\tfrac{1}{2}-\beta) \\ &P_1(\zeta) = \zeta - 1 \end{aligned} \tag{3.41}$$

the characteristic equation is

$$P^{\mathbf{q}}_{\omega h}(\zeta) = \det \begin{bmatrix} P_{\alpha_f}(\zeta)\omega^2 & 0 & P_{\alpha_m}(\zeta) \\ P_1(\zeta) & -h & -P_\beta(\zeta)h^2 \\ 0 & P_1(\zeta) & -P_\gamma(\zeta)h \end{bmatrix} = 0 \tag{3.42}$$

which is equivalent to:

$$P^{\mathbf{q}}_{\omega h}(\zeta) = P_{\alpha_f}(P_\gamma + P_1\,P_\beta)\,(\omega h)^2 + P_1^2\,P_{\alpha_m} = 0 \tag{3.43}$$

The algorithm is stable if all the roots of $P^{\mathbf{q}}_{\omega h}(\zeta) = 0$ are inside the unit circle. Unconditional stability requires this property to be satisfied whatever the value of ωh. Therefore, an important property of the algorithm is the spectral radius associated with the roots $\zeta_i^{\mathbf{q}}$ of $P^{\mathbf{q}}_{\omega h}$ for $\omega h \to \infty$:

$$\rho^{\mathbf{q}}_\infty = \max_i |\zeta_i^{\mathbf{q}}| \tag{3.44}$$

which should be less than one for unconditional stability. This spectral radius is computed from the characteristic equation:

$$P^{\mathbf{q}}_\infty = P_{\alpha_f}(P_\gamma + P_1\,P_\beta) = 0 \tag{3.45}$$

If we consider the optimal parameters β and γ given by (3.31), this equation has one simple root and one double root:

$$\zeta_1^{\mathbf{q}} = \frac{-\alpha_f}{1-\alpha_f} \qquad \zeta_{2,3}^{\mathbf{q}} = -\frac{1-\alpha_{fm}}{1+\alpha_{fm}} \tag{3.46}$$

$\zeta_1^{\mathbf{q}}$ is the so-called spurious root, and $\zeta_{2,3}^{\mathbf{q}}$ are the principal roots. A usual design requirement is $|\zeta_1^{\mathbf{q}}| \leq |\zeta_{2,3}^{\mathbf{q}}|$. The condition $|\zeta_1^{\mathbf{q}}| = |\zeta_{2,3}^{\mathbf{q}}| = \rho_\infty^{\mathbf{q}}$ leads to the optimal formulae (3.30) proposed by Chung and Hulbert [CH93], and the linear stability is then guaranteed by the design constraint $\rho_\infty^{\mathbf{q}} < 1$. It is easy to verify that the Hilber-Hughes-Taylor algorithm has a spectral radius less than one.

Constrained system

The linearized equations of an undamped system are:

$$\begin{aligned} \mathbf{M}\,\ddot{\mathbf{q}} + \mathbf{K}\,\mathbf{q} + \mathbf{B}^T\,\boldsymbol{\lambda} &= \mathbf{0} \\ \mathbf{B}\,\mathbf{q} &= \mathbf{0} \end{aligned} \tag{3.47}$$

As pointed out by Géradin and Cardona [GC01], this system is not diagonalizable due to the presence of the constraints. However, these authors demonstrated that it could be transformed into a canonical form by a linear transformation:

Theorem 3.1 *If* $\mathbf{K}$ *is symmetric positive semi-definite,* $\mathbf{M}$ *is symmetric positive definite, and* $\mathbf{B}$ *has full rank, there exists a transformation matrix* $\mathbf{T}_m$ *such that*

$$\mathbf{T}_m^T \begin{bmatrix} \mathbf{K} & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \mathbf{T}_m = \begin{bmatrix} \boldsymbol{\Omega}^2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}^{cc} \\ \mathbf{0} & \mathbf{I}^{cc} & \mathbf{0} \end{bmatrix} \tag{3.48}$$

$$\mathbf{T}_m^T \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{T}_m = \begin{bmatrix} \mathbf{I}^{rr} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}^{cc} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{3.49}$$

where $\boldsymbol{\Omega}^2$ *is a* $(n-m)\times(n-m)$ *diagonal matrix,* $\mathbf{I}^{cc}$ *is the identity matrix of dimension* $m \times m$, $\mathbf{I}^{rr}$ *is the identity matrix of dimension* $(n-m)\times(n-m)$, *and* $\mathbf{0}$ *are null matrices with appropriate dimensions.*

This transformation matrix $\mathbf{T}_m$ leads to the definition of a new set of mechanical variables $\mathbf{q}^r$, $\mathbf{q}^c$, and $\mathbf{q}^\lambda$:

$$\begin{bmatrix} \mathbf{q} \\ \boldsymbol{\lambda} \end{bmatrix} = \mathbf{T}_m \begin{bmatrix} \mathbf{q}^r \\ \mathbf{q}^c \\ \mathbf{q}^\lambda \end{bmatrix} = \begin{bmatrix} \mathbf{T}_{\mathbf{q}r} & \mathbf{T}_{\mathbf{q}c} & \mathbf{0} \\ \mathbf{T}_{\boldsymbol{\lambda}r} & \mathbf{T}_{\boldsymbol{\lambda}c} & \mathbf{T}_{\boldsymbol{\lambda}\lambda} \end{bmatrix} \begin{bmatrix} \mathbf{q}^r \\ \mathbf{q}^c \\ \mathbf{q}^\lambda \end{bmatrix} \tag{3.50}$$

$\mathbf{q}^c$ are m constrained dofs, $\mathbf{q}^\lambda$ are m multipliers-like dofs, and $\mathbf{q}^r$ are $n-m$ independent dofs. Actually, those names do not reflect the true complexity of the transformation, since the Lagrange multipliers $\boldsymbol{\lambda}$ are also connected with $\mathbf{q}^r$ and $\mathbf{q}^c$.

After transformation, the set of dynamic equations becomes:

$$\begin{aligned} \ddot{\mathbf{q}}^r + \boldsymbol{\Omega}^2\, \mathbf{q}^r &= \mathbf{0} \\ \ddot{\mathbf{q}}^c + \mathbf{q}^\lambda &= \mathbf{0} \\ \mathbf{q}^c &= \mathbf{0} \end{aligned} \tag{3.51}$$

Following the procedure detailed for the unconstrained case, the characteristic equation is the determinant of the following matrix:

$$\left[\begin{array}{ccc|ccc|ccc} P_{\alpha_f}\boldsymbol{\Omega}^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & P_{\alpha_m}\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & P_{\alpha_f}\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & P_{\alpha_m}\mathbf{I} & \mathbf{0} \\ \mathbf{0} & P_{\alpha_f}\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline P_1\mathbf{I} & \mathbf{0} & \mathbf{0} & -h\,\mathbf{I} & \mathbf{0} & \mathbf{0} & -P_\beta h^2\,\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & P_1\mathbf{I} & \mathbf{0} & \mathbf{0} & -h\,\mathbf{I} & \mathbf{0} & \mathbf{0} & -P_\beta h^2\,\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & P_1\mathbf{I} & \mathbf{0} & \mathbf{0} & -h\,\mathbf{I} & \mathbf{0} & \mathbf{0} & -P_\beta h^2\,\mathbf{I} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & P_1\mathbf{I} & \mathbf{0} & \mathbf{0} & -P_\gamma h\,\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & P_1\mathbf{I} & \mathbf{0} & \mathbf{0} & -P_\gamma h\,\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & P_1\mathbf{I} & \mathbf{0} & \mathbf{0} & -P_\gamma h\,\mathbf{I} \end{array}\right] \tag{3.52}$$

After lengthy but straightforward algebraic manipulations, and referring to the characteristic polynomial $P^{\mathbf{q}}_{\omega h}$ associated with the unconstrained problem, one finds that the characteristic equation is equivalent to:

$$\left(P^{\mathbf{q}}_{\infty}(\zeta)\right)^{2m} \det \begin{bmatrix} P_{\alpha_f}(\zeta)\,\boldsymbol{\Omega}^2 & \mathbf{0} & P_{\alpha_m}(\zeta)\,\mathbf{I} \\ P_1(\zeta)\mathbf{I} & -h\,\mathbf{I} & -P_\beta(\zeta)h^2\,\mathbf{I} \\ \mathbf{0} & P_1(\zeta)\mathbf{I} & -P_\gamma(\zeta)h\,\mathbf{I} \end{bmatrix} = 0 \tag{3.53}$$

The second factor can be interpreted as the characteristic equation when the algorithm is applied to an unconstrained diagonal system, so that equation (3.53) can be restated:

$$(P_\infty^{\mathbf{q}}(\zeta))^{2m} \prod_i P_{\omega_i h}^{\mathbf{q}}(\zeta) = 0 \tag{3.54}$$

where ω_i are the diagonal terms of $\mathbf{\Omega}$. Any root of this equation is either a root of the polynomial $P_{\omega_i h}^{\mathbf{q}}$ obtained in the unconstrained case, or a root of the same polynomial at infinite frequency. We conclude that if the generalized-α algorithm leads to a stable integration of the unconstrained problem, and if the spectral radius at infinity $\rho_\infty^{\mathbf{q}}$ is smaller than 1, the global stability is guaranteed. If $\rho_\infty^{\mathbf{q}} = 1$, it can be demonstrated that the numerical integration is weakly unstable [GC01].

3.3.4 Convergence analysis

This section investigates the second-order accuracy of the generalized-α algorithm, and as for the linear stability, the analysis starts with the unconstrained case.

Local truncation error: the unconstrained case

If the mass matrix is constant and non-singular, it is equivalent to study the behavior of the algorithm applied to the system:

$$\ddot{\mathbf{q}} = \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, t) \tag{3.55}$$

According to the generalized-α algorithm, the discrete-time equation is

$$(1 - \alpha_m)\, \ddot{\mathbf{q}}_{n+1} + \alpha_m\, \ddot{\mathbf{q}}_n = (1 - \alpha_f)\, \mathbf{f}_{n+1} + \alpha_f\, \mathbf{f}_n \tag{3.56}$$

In this work, $\mathbf{q}_{n+1}, \dot{\mathbf{q}}_{n+1}$, and $\ddot{\mathbf{q}}_{n+1}$ conventionally denote the *numerical solutions*, whereas $\mathbf{q}(t_{n+1}), \dot{\mathbf{q}}(t_{n+1})$, and $\ddot{\mathbf{q}}(t_{n+1})$ refer to the *exact solutions*. The numerical solutions $\mathbf{q}_{n+1}, \dot{\mathbf{q}}_{n+1}$ satisfy the Newmark formulae (3.22), and the differences with the exact solutions $\mathbf{q}(t_{n+1}), \dot{\mathbf{q}}(t_{n+1})$ are the local truncation errors:

$$\mathbf{e}_{n+1} = \mathbf{q}(t_{n+1}) - \mathbf{q}_{n+1} \tag{3.57}$$

$$\mathbf{e}'_{n+1} = \dot{\mathbf{q}}(t_{n+1}) - \dot{\mathbf{q}}_{n+1} \tag{3.58}$$

The order 2 condition is:

$$\mathbf{e}_{n+1} = \mathcal{O}(h^3) \tag{3.59}$$

$$\mathbf{e}'_{n+1} = \mathcal{O}(h^3) \tag{3.60}$$

In order to exploit the theory of linear multistep algorithms, it is useful to note that the one-step generalized-α method hides a two-step linear formula for the velocities, and a three-step linear formula for the displacements [EBB02]. Indeed, from the integration formulae (3.22) and (3.56) developed at time t_{n+1} and t_{n+2}, the elimination of the accelerations $\ddot{\mathbf{q}}_n$, $\ddot{\mathbf{q}}_{n+1}$ and $\ddot{\mathbf{q}}_{n+2}$ leads to the two-step formulae:

$$\sum_{k=0}^{2} \alpha_k \, \mathbf{q}_{n+k} + \sum_{k=0}^{1} h \, \mu_k \, \dot{\mathbf{q}}_{n+k} = \sum_{k=0}^{2} h^2 \, \beta_k \, \mathbf{f}_{n+k} \tag{3.61}$$

$$\sum_{k=0}^{2} \alpha_k \, \dot{\mathbf{q}}_{n+k} = \sum_{k=0}^{2} h \, \gamma_k \, \mathbf{f}_{n+k} \tag{3.62}$$

with

$$\begin{array}{lll} \alpha_0 = -\alpha_m, & \alpha_1 = -1 + 2\alpha_m, & \alpha_2 = 1 - \alpha_m, \\ \mu_0 = -\alpha_m, & \mu_1 = -1 + \alpha_m, & \\ \beta_0 = (1/2 - \beta)\,\alpha_f, & \beta_1 = 1/2 - \beta - \alpha_f/2 + 2\,\beta\,\alpha_f, & \beta_2 = \beta\,(1 - \alpha_f), \\ \gamma_0 = (1 - \gamma)\,\alpha_f, & \gamma_1 = 1 - \gamma - \alpha_f + 2\,\gamma\,\alpha_f, & \gamma_2 = \gamma\,(1 - \alpha_f). \end{array} \tag{3.63}$$

and a further elimination of the velocities in the first equation would lead to a three-step formula for the displacements. Equation (3.62) is a good basis for the analysis of the local truncation error *at the velocity level*. Since it is a standard multistep formula, the order 2 condition can be found in classical textbooks such as Hairer *et al.* [HNW87]:

$$\alpha_0 + \alpha_1 + \alpha_2 = 0 \tag{3.64}$$

$$\alpha_1 + 2\,\alpha_2 = \gamma_0 + \gamma_1 + \gamma_2 \tag{3.65}$$

$$\alpha_1 + 4\,\alpha_2 = 2\,\gamma_1 + 4\,\gamma_2 \tag{3.66}$$

These relations come from the substitution of $\dot{\mathbf{q}}$ and $\mathbf{f}$ by their Taylor series expansion in the expression of the local error. The first two equations are automatically satisfied by the parameters given in (3.63), whereas the last equation yields

$$\gamma = \frac{1}{2} + \alpha_f - \alpha_m \tag{3.67}$$

which is the order 2 condition for the generalized-α method (3.31).

A the displacement level, the order condition is obtained by a similar analysis of the three-step formulation. For conciseness, the demonstration is not reproduced here; it has been developed by Erlicher *et al.* [EBB02].

Those results rely on the initial assumption of a constant mass matrix. In flexible multibody dynamics, the mass matrix is actually varying between t_n and t_{n+1}, so that the discretized equation becomes

$$(1-\alpha_m)\,\mathbf{M}_{n+1}\,\ddot{\mathbf{q}}_{n+1} + \alpha_m\,\mathbf{M}_n\,\ddot{\mathbf{q}}_n = (1-\alpha_f)\,\mathbf{g}_{n+1} + \alpha_f\,\mathbf{g}_n \tag{3.68}$$

which is not equivalent to equation (3.56). However, a deeper analysis of the updated Lagrangian formulation shows that the variation of the mass matrix over one time-step is usually of order 2, leading to $\mathcal{O}(h^2)$, $\mathcal{O}(h^3)$ and $\mathcal{O}(h^4)$ local truncation error in $\ddot{\mathbf{q}}_{n+1}$, $\dot{\mathbf{q}}_{n+1}$ and $\mathbf{q}_{n+1}$, respectively. Therefore, the order 2 condition is usually observed for the displacements and velocities.

Global convergence: the unconstrained case

The local truncation error analysis gives some information about the error over one time-step. However, when a finite time-interval is considered, the propagation of the errors may jeopardize the convergence of the numerical solution.

According to Hairer *et al.* [HNW87], a linear multistep algorithm is convergent of order p if it is stable and if its local truncation error is of order p. In particular, let us consider the $n_{\mathbf{x}}$ dimensional first-order ODE:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t) \tag{3.69}$$

solved using a k-step stable method, whose local truncation error satisfies the order p condition

$$\|\mathbf{x}(t_{n+1}) - \mathbf{x}_{n+1}\| \leq M\,h^{p+1} \tag{3.70}$$

We also define the $(k\,n_{\mathbf{x}}) \times 1$ vectors collecting the coordinate vectors of the last k steps:

$$\mathbf{X}_i^T = (\mathbf{x}_{i+k-1}^T, ..., \mathbf{x}_i^T) \tag{3.71}$$

$$\mathbf{X}(t_i)^T = (\mathbf{x}(t_{i+k-1})^T, ..., \mathbf{x}(t_i)^T) \tag{3.72}$$

The following bound on the global error after n time-steps is given by Hairer *et al.* [HNW87]

$$\|\mathbf{X}(t_n) - \mathbf{X}_n\| \leq \|\mathbf{X}(t_0) - \mathbf{X}_0\| \, e^{nhL^*} + \frac{M \, h^p}{L^*} \left(e^{nhL^*} - 1\right) \tag{3.73}$$

where L^* is a Lipschitz constant. An initial error in the initial conditions is at most amplified by a coefficient depending exponentially on the length of the time interval nh. Therefore, it is especially important to limit this phenomenon by the definition of *consistent initial conditions*.

Since the generalized-α method hides a multistep algorithm, the error propagation might be characterized by a similar property, and the consistency of the initial conditions is of practical interest to guarantee the global convergence. For a multistep method, the initial conditions are given as the vector $\mathbf{X}_0$ of the solution at the k first steps, which is necessary to initiate the integration algorithm. However, in its pseudo-one-step formulation, the generalized-α algorithm starts on the basis of the values $\mathbf{q}_0$, $\dot{\mathbf{q}}_0$ and $\ddot{\mathbf{q}}_0$. For full consistency, $\ddot{\mathbf{q}}_0$ should satisfy:

$$(1 - \alpha_m) \, \ddot{\mathbf{q}}_0 + \alpha_m \, \ddot{\mathbf{q}}_{-1} = (1 - \alpha_f) \, \mathbf{f}_0 + \alpha_f \, \mathbf{f}_{-1} \tag{3.74}$$

In practice, this equation is helpless for the determination of $\ddot{\mathbf{q}}_0$, and it is usually replaced by the simplified consistency condition:

$$\ddot{\mathbf{q}}_0 = \mathbf{f}_0 \tag{3.75}$$

This expression satisfies (3.74) with an $\mathcal{O}(h)$ error. From a dimensional analysis, this initial error at the acceleration level will contaminate the displacements and velocities during the first steps with $\mathcal{O}(h^3)$ and $\mathcal{O}(h^2)$ errors, respectively. This error in the initial steps will be coupled, propagated and amplified by the integration procedure, leading to a maximal global error of $\mathcal{O}(h^2)$, so that second-order convergence is still guaranteed. However, if the simplified consistency condition (3.75) is not satisfied, the error in the initial values may drop one order, with disastrous consequences on the global convergence of the results.

The constrained case

The previous convergence results can be extended for constrained systems. In Figure 3.2, the time-integration algorithm developed for flexible multibody

systems guarantees that the algebraic constraints:

$$\mathbf{\Phi}(\mathbf{q}) = \mathbf{0} \tag{3.76}$$

are satisfied at each time-step, if they were satisfied by the initial conditions. Hidden constraints should also be satisfied by the dynamic variables at all differentiated levels, and in particular, at the velocity and acceleration levels:

$$\dot{\mathbf{\Phi}} = \mathbf{\Phi_q}\ \dot{\mathbf{q}} = \mathbf{0} \tag{3.77}$$

$$\ddot{\mathbf{\Phi}} = \dot{\mathbf{\Phi}}_\mathbf{q}\ \dot{\mathbf{q}} + \mathbf{\Phi_q}\ \ddot{\mathbf{q}} = \mathbf{0} \tag{3.78}$$

Those equations are, however, not considered in the algorithm, which might lead to significant errors.

In the special case of linear constraints, $\mathbf{\Phi_q}$ is constant, and the constraints at the acceleration level become:

$$\mathbf{\Phi_q}\ \ddot{\mathbf{q}} = \mathbf{0} \tag{3.79}$$

After inspection of the predictor and corrector steps of the numerical algorithm, the constraints at displacement, velocity and acceleration levels are satisfied at t_{n+1} if they were initially satisfied at t_n. Therefore, the local truncation error is of order 2, as in the unconstrained case.

Since the constraints are satisfied at each time grid point, the constraint violation of the numerical solution is propagated at a very high frequency which depends on the step size. In agreement with the stability analysis, the amplification factor of this phenomenon is associated with the spectral radius of the integration algorithm at high frequencies. Therefore, the numerical damping is responsible for the decay of this error.

In case of nonlinear but smooth constraints, if we do not consider nonlinear amplification effects, the violation of the hidden constraints resulting from the nonlinearity may be seen as a disturbance, which is filtered by the algorithm in the same manner. In this sense, those errors do not accumulate throughout the integration process, and the convergence results demonstrated for unconstrained systems are still relevant.

From a practical perspective, the computation of consistent initial velocities and accelerations may rely on equations (3.77) and (3.78) (or its linearized counterpart (3.79)).

3.4 Implementation

The concepts presented in this chapter have been implemented in the Mecano program of the Samcef software [SAM99]. In 2005, this program has reached industrial maturity for several years, it is well documented and it contains a huge library of finite elements and types of analysis, which is very convenient for the user. Mecano has been extensively exploited for problems in aerospace, robotics, automotive engineering and machine-tools.

However, for the developments realized in this research, the Oofelie multi-physics platform [CKG94] has been selected for several reasons. Oofelie is an acronym for Object Oriented Finite Element Led by Interactive Executor. It is written in C++, and modern programming concepts are favourable for an open architecture, well-suited for new developments. Its kernel has been developed to deal with strongly coupled multiphysics problems, with great care about the modularity and efficiency issues. Since the developments related with flexible multibody dynamics only started a couple of years ago, a restricted library of the most significant elements and algorithms is currently available in Oofelie. Nevertheless, those capabilities are sufficient to demonstrate the relevance and the efficiency of the innovative concepts developed in this research.

IV

Integrated Simulation of Mechatronic Systems

In the introduction, the motivation for an integrated simulation tool able to predict the dynamic behavior of a mechatronic system was clearly demonstrated. The analysis should account for all the technological components: the mechanism, the control system, the actuators, the sensors, etc. We have also seen that the Finite Element method is a general and reliable modeling approach able to deal efficiently with complex flexible mechanisms.

This chapter presents some extensions of the Finite Element method for the simulation of mechatronic systems. The description of non-mechanical elements within a Finite Element context is not an innovative idea, and for instance, user-elements are available in most industrial software for multibody systems. However, the implementation of a user-element is an intricate, time-consuming and sometimes unreliable process, which motivates a more systematic and theoretically founded approach. Hence, we propose to integrate the *block diagram language* within the Finite Element framework, as we have suggested in [BDG04b]. Relying on a nonlinear state-space description, this approach presents the advantages of modularity, generality, with a language very familiar to control engineers.

The methodology developed here is illustrated in Figure 4.1. The mechanical system is modeled using the Finite Element formalism for flexible multibody systems presented in chapter 3, whereas the control system is described using the block diagram language. The strongly coupled mechanical and state equations are obtained by numerical assembly, and their time-domain simulation is based

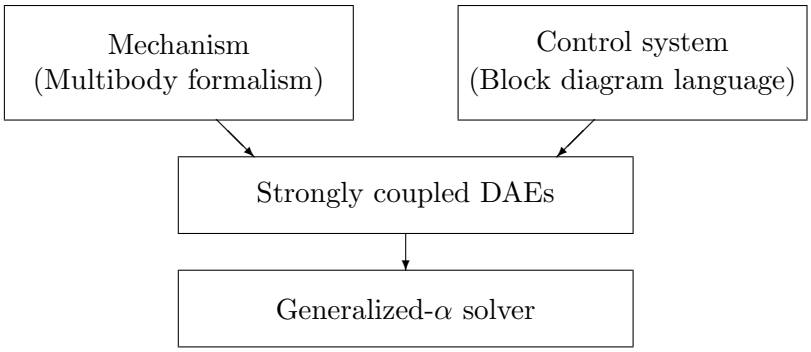


Figure 4.1: Strongly coupled and modular simulation.

on an extension of the generalized-α method.

In a mechatronic system, discontinuous phenomena may occur, *e.g.* due to the sampling effect in a digital electronic system or to the saturation of a component. From a numerical point of view, a discontinuous transition requires a computation restart procedure, which is also considered here.

The organization of the chapter is as follows. First, a state-space formalism for the simulation of control systems is presented in section 4.1, with convergence and stability results. Section 4.2 describes the integrated simulation method and analyzes its stability properties. A specific treatment for systems with discontinuous phenomena is developed in section 4.3. After the presentation of a few implementation issues in section 4.4, three examples with increasing complexity illustrate the method: a four-bar mechanism, a Scara robot, and a vehicle semi-active suspension.

4.1 Simulation of control systems

Figure 4.2 illustrates the interactions between a mechanism and a continuous control system. The control system includes all the non-mechanical elements, such as the controller, the actuators and the sensors. Mathematically, the interactions are represented by two vectors: the $n_m \times 1$ vector $\mathbf{w}^m$ contains the sensor measurements exploited by the control system and any other mechanical variables that may influence its dynamic behavior, whereas the $n_a \times 1$ vector $\mathbf{g}^a$ represents the generalized forces exerted by the actuators on the mechanism.

This section focuses on the description of the control system using the block

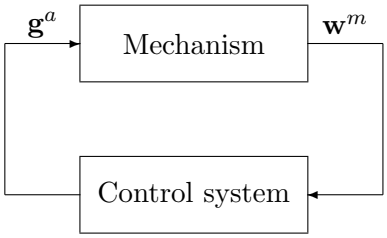


Figure 4.2: Interactions between a mechanism and a control system.

diagram language, and on the simulation of the resulting state-space model. The simulation of the coupled system will be considered later, in section 4.2.

4.1.1 State equations

The most general formalism for the description of the control system is certainly the descriptor state-space format:

$$\begin{aligned} \mathbf{f}^{s*}(\mathbf{w}^m, \mathbf{x}, \dot{\mathbf{x}}, t) &= \mathbf{0} \\ \mathbf{f}^{o*}(\mathbf{w}^m, \mathbf{x}, \mathbf{g}^a, t) &= \mathbf{0} \end{aligned} \tag{4.1}$$

where the $n_{\mathbf{x}} \times 1$ vector $\mathbf{x}$ represents the state variables, $\mathbf{f}^{s*}$ are the $n_{\mathbf{x}}$ differential equations of the dynamic states, whereas $\mathbf{f}^{o*}$ are the n_a algebraic output equations. Control engineers are more familiar with the explicit state-space format, which is a special case of the descriptor state-space format:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}^s(\mathbf{w}^m, \mathbf{x}, t) \\ \mathbf{g}^a &= \mathbf{f}^o(\mathbf{w}^m, \mathbf{x}, t) \end{aligned} \tag{4.2}$$

In general, the transformation from (4.1) to (4.2), is not always possible nor trivial.

Such a global input/output black box description is compact and efficient. However, a functional decomposition into subsystems may lead to an advantageous modular approach, as illustrated in Figure 4.3. At the subsystem level, the explicit state-space equations can be formulated more easily, and we consider that each element e is characterized by explicit state equations, with respect to its own inputs $\mathbf{u}^{(e)}$ and outputs $\mathbf{y}^{(e)}$:

$$\begin{aligned} \dot{\mathbf{x}}^{(e)} &= \mathbf{f}^{s(e)}(\mathbf{u}^{(e)}, \mathbf{x}^{(e)}, t) \\ \mathbf{y}^{(e)} &= \mathbf{f}^{o(e)}(\mathbf{u}^{(e)}, \mathbf{x}^{(e)}, t) \end{aligned} \tag{4.3}$$

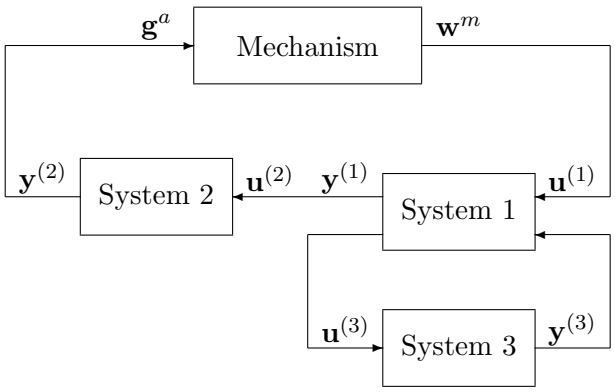


Figure 4.3: Modular approach for the description of the control system.

Hence, one defines the global input and output vectors:

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}^{(1)} \\ \vdots \\ \mathbf{u}^{(n_e)} \end{bmatrix}, \qquad \mathbf{y} = \begin{bmatrix} \mathbf{y}^{(1)} \\ \vdots \\ \mathbf{y}^{(n_e)} \end{bmatrix} \tag{4.4}$$

with the connectivity relations:

$$\mathbf{u} = \mathbf{L}^{im}\, \mathbf{w}^m + \mathbf{L}^{io}\, \mathbf{y} \qquad \text{and} \qquad \mathbf{g}^a = \mathbf{L}^{ao}\, \mathbf{y} \tag{4.5}$$

where $\mathbf{L}^{im}$, $\mathbf{L}^{io}$ and $\mathbf{L}^{ao}$ are boolean localization matrices.

The assembled equations of the block diagram model are summarized by:

$$\dot{\mathbf{x}} = \mathbf{f}^s(\mathbf{u}, \mathbf{x}, t) \tag{4.6}$$

$$\mathbf{y} = \mathbf{f}^o(\mathbf{u}, \mathbf{x}, t) \tag{4.7}$$

$$\mathbf{u} = \mathbf{L}^{im}\, \mathbf{w}^m + \mathbf{L}^{io}\, \mathbf{y} \tag{4.8}$$

with the actuator outputs:

$$\mathbf{g}^a = \mathbf{L}^{ao}\, \mathbf{y} \tag{4.9}$$

$\mathbf{u}$ and $\mathbf{g}^a$ are explicitly defined by boolean equations; in the numerical code, they are not implemented as independent variables. However, they appear in the formal presentation for notational convenience.

Those equations are equivalent to the descriptor state-space equations (4.1), and we observe that the transformation to the explicit form (4.2) would require the

elimination of $\mathbf{u}$ and $\mathbf{y}$, using an algebraic transformation. This means that even though we restrict the subsystems description to the ODEs (4.3), the assembled equations are DAEs, where the algebraic equations come from the input/output connections between subsystems. The theoretical results of the time-integration scheme will rely on an additional technical assumption on the algebraic structure of those DAEs, which is presented hereafter.

4.1.2 Semi-explicit index-1 assumption

In the general theory of DAEs [BCP96], a semi-explicit index-1 system is written as $n_{\mathbf{x}}$ state equations and $n_{\mathbf{z}}$ constraint equations:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{z}, t) \tag{4.10}$$

$$\mathbf{0} = \mathbf{g}(\mathbf{x}, \mathbf{z}, t) \tag{4.11}$$

with the property that the $n_{\mathbf{z}} \times n_{\mathbf{z}}$ Jacobian $\mathbf{g}_{\mathbf{z}}$ exists and is invertible in the neighborhood of the exact solution. From the implicit function theorem, $\mathbf{g}$ can thus be solved for the algebraic variables $\mathbf{z}$ in terms of $\mathbf{x}$ and t (*i.e.*, the function $\mathbf{z} = \widetilde{\mathbf{g}}(\mathbf{x}, t)$ can be defined locally). Hence, a very natural implementation of a time-integration scheme is to require the constraint (4.11) to be satisfied by the algebraic variables $\mathbf{z}$ at each time-step, while applying a standard multistep formula to (4.10) for the time-integration of the dynamic variables $\mathbf{x}$. Indeed, the algorithm behaves as if it were applied to the equivalent ODE:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \widetilde{\mathbf{g}}(\mathbf{x}, t), t) \tag{4.12}$$

A multistep integrator applied to semi-explicit index-1 DAEs in this manner is thus stable and convergent to the same order of accuracy as if it were applied to standard stiff ODEs.

In our case, the input and output variables are the algebraic variables, and the block diagram model (4.6), (4.7) and (4.8) is a semi-explicit index-1 system if the Jacobian satisfies the following regularity condition.

Assumption 4.1

$$\det \begin{bmatrix} \mathbf{I} & -\mathbf{f}_{\mathbf{u}}^{o} \\ -\mathbf{L}^{io} & \mathbf{I} \end{bmatrix} \neq 0 \tag{4.13}$$

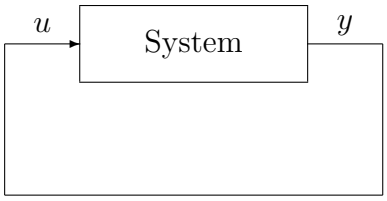


Figure 4.4: A trivial system.

In this case, the implicit function theorem can be invoked to solve (4.7) and (4.8) for the inputs and outputs:

$$\mathbf{y} = \widetilde{\mathbf{f}}^{o}(\mathbf{w}^{m}, \mathbf{x}, t) \tag{4.14}$$

$$\mathbf{u} = \widetilde{\mathbf{f}}^{u}(\mathbf{w}^{m}, \mathbf{x}, t) \tag{4.15}$$

and the numerical algorithm behaves as if it were applied to the equivalent ODE:

$$\dot{\mathbf{x}} = \widetilde{\mathbf{f}}^{s}(\mathbf{w}^{m}, \mathbf{x}, t) \tag{4.16}$$

with $\widetilde{\mathbf{f}}^{s}(\mathbf{w}^{m}, \mathbf{x}, t) = \mathbf{f}^{s}(\widetilde{\mathbf{f}}^{u}(\mathbf{w}^{m}, \mathbf{x}, t), \mathbf{x}, t)$.

Assumption 4.1 is not satisfied if the block diagram model hides algebraic constraints for the states $\mathbf{x}$ and/or for the mechanical inputs $\mathbf{w}^{m}$. This pathological situation can be illustrated with the trivial system described by the scalar linear equations (Figure 4.1.2):

$$\begin{aligned} \dot{x} &= a\,x + b\,u \\ y &= c\,x + u \\ u &= y \end{aligned} \tag{4.17}$$

The input/output equations hide the constraint $x = 0$, so that the output variable $y = u$ plays the role of a Lagrange multiplier in the dynamic equation. The block diagram formalism presented here is not really appropriate for such a higher-index DAE. In the following, Assumption 4.1 is supposed to be satisfied, so that an efficient strategy can be developed with guaranteed reliability.

4.1.3 Time-integration algorithm

This section presents a methodology for the simulation of a block diagram model *isolated from the mechanical system*. Therefore, we consider that the measurements are known functions of time $\mathbf{w}^{m} = \mathbf{w}^{m}(t)$.

A simple and interesting integrator for first-order ODEs is the trapezoidal rule:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \frac{h}{2}\left(\dot{\mathbf{x}}_n + \dot{\mathbf{x}}_{n+1}\right) \tag{4.18}$$

This formula is second-order accurate and A-stable, but it is not appropriate for stiff problems, due to the absence of numerical filtering at high frequencies [HW91]. An alternative, stabilized scheme is given by

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h\,(1-\theta)\,\dot{\mathbf{x}}_n + h\,\theta\,\dot{\mathbf{x}}_{n+1} \qquad \theta \in [1/2, 1] \tag{4.19}$$

with only first-order accuracy for $\theta \neq \frac{1}{2}$.

In order to introduce high-frequency numerical damping without tremendous accuracy loss, improved integration formula are necessary, such as multistep or Runge-Kutta methods [HW91]. The default of the one-step formula (4.19) is similar to the default of the original Newmark formulae. Introducing the modified residual equation, Hilber, Hughes and Taylor [HHT77], followed by Chung and Hulbert [CH93], proposed an elegant generalization of the Newmark scheme as a three-step method, with an implementation very close to a one-step method. The resulting generalized-α method was presented in section 3.3.2. Jansen *et al.* [JWH00] proposed an extension of this method for first-order differential equations in fluid dynamics.

We propose to apply a similar strategy here, which suggests the modified residual equation

$$(1-\delta_m)\,\dot{\mathbf{x}}_{n+1} + \delta_m\,\dot{\mathbf{x}}_n - (1-\delta_f)\,\mathbf{f}^s_{n+1} - \delta_f\,\mathbf{f}^s_n = \mathbf{0} \tag{4.20}$$

$$(1-\delta_f)\,(\mathbf{y}_{n+1} - \mathbf{f}^o_{n+1}) + \delta_f\,(\mathbf{y}_n - \mathbf{f}^o_n) = \mathbf{0} \tag{4.21}$$

where δ_m and δ_f are numerical parameters of the method. The modification of the output equation does not alter the local error on the output variables. A perfect analogy can be observed between the treatment of the state variables in this method and the treatment of the velocities in the generalized-α method. Defining $\delta_{fm} = \delta_f - \delta_m$, we immediately conclude that second-order accuracy is obtained for

$$\theta = \frac{1}{2} + \delta_{fm} \tag{4.22}$$

According to section 3.3.4, consistent initial conditions should satisfy:

$$\dot{\mathbf{x}}_0 = \mathbf{f}^s_0 \tag{4.23}$$

$$\mathbf{y}_0 = \mathbf{f}^o_0 \tag{4.24}$$

leading to the propagation of an $\mathcal{O}(h^2)$ initial error on the states.

The integration scheme follows a predictor-corrector procedure. The prediction formula is obtained with the zero state rates assumption:

$$\begin{aligned} \dot{\mathbf{x}}^0_{n+1} &= 0 \\ \mathbf{x}^0_{n+1} &= \mathbf{x}_n + (1-\theta)\, h\, \dot{\mathbf{x}}_n \\ \mathbf{y}^0_{n+1} &= \mathbf{y}_n \end{aligned} \tag{4.25}$$

and the correction of the state rates follows:

$$\triangle \dot{\mathbf{x}}_{n+1} = \frac{1}{\theta\, h}\, \triangle \mathbf{x}_{n+1} \tag{4.26}$$

The linearized form of the discretized state equations (4.20), combined with the linearized input/output relations are:

$$(1-\delta_m)\, \Delta\dot{\mathbf{x}} - (1-\delta_f)\, \mathbf{f}^s_{\mathbf{x}}\, \Delta\mathbf{x} - (1-\delta_f)\, \mathbf{f}^s_{\mathbf{u}}\, \Delta\mathbf{u} = -\mathbf{res}^s_k \tag{4.27}$$

$$(1-\delta_f)\, (\Delta\mathbf{y} - \mathbf{f}^o_{\mathbf{x}}\, \Delta\mathbf{x} - \mathbf{f}^o_{\mathbf{u}}\, \Delta\mathbf{u}) = -\mathbf{res}^o_k \tag{4.28}$$

$$\Delta\mathbf{u} = \mathbf{L}^{io}\, \Delta\mathbf{y} \tag{4.29}$$

which can be put in matrix form using (4.26), after elimination of $\Delta\mathbf{u}$ and $\Delta\dot{\mathbf{x}}$:

$$\left((1-\delta_m)\, \frac{1}{\theta\, h} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} + (1-\delta_f) \begin{bmatrix} -\mathbf{f}^s_{\mathbf{x}} & -\mathbf{f}^s_{\mathbf{u}}\, \mathbf{L}^{io} \\ -\mathbf{f}^o_{\mathbf{x}} & \mathbf{I} - \mathbf{f}^o_{\mathbf{u}}\, \mathbf{L}^{io} \end{bmatrix} \right) \begin{bmatrix} \Delta\mathbf{x} \\ \Delta\mathbf{y} \end{bmatrix} = \begin{bmatrix} -\mathbf{res}^s_k \\ -\mathbf{res}^o_k \end{bmatrix} \tag{4.30}$$

In chapter 3, the linear stability analysis of the generalized-α method was investigated for second-order undamped equations. We may reproduce this analysis for first-order equations.

4.1.4 Linear stability analysis

In order to assess the stability properties of the algorithm in the linear regime, let us consider the linearized equations, obtained as in the previous paragraph:

$$\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{y}} \end{bmatrix} + \begin{bmatrix} -\mathbf{f}^s_{\mathbf{x}} & -\mathbf{f}^s_{\mathbf{u}}\, \mathbf{L}^{io} \\ -\mathbf{f}^o_{\mathbf{x}} & \mathbf{I} - \mathbf{f}^o_{\mathbf{u}}\, \mathbf{L}^{io} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \mathbf{0} \tag{4.31}$$

Using Assumption 4.1, this DAE system is equivalent to an ODE system supplemented with an algebraic output equation. Indeed, since $\mathbf{I} - \mathbf{f}^o_{\mathbf{u}}\, \mathbf{L}^{io}$ is not singular, it is possible to extract explicitly the algebraic outputs:

$$\mathbf{y} + \mathbf{C}\, \mathbf{x} = \mathbf{0} \qquad \text{with} \qquad \mathbf{C} = -\left(\mathbf{I} - \mathbf{f}^o_{\mathbf{u}}\, \mathbf{L}^{io}\right)^{-1}\, \mathbf{f}^o_{\mathbf{x}} \tag{4.32}$$

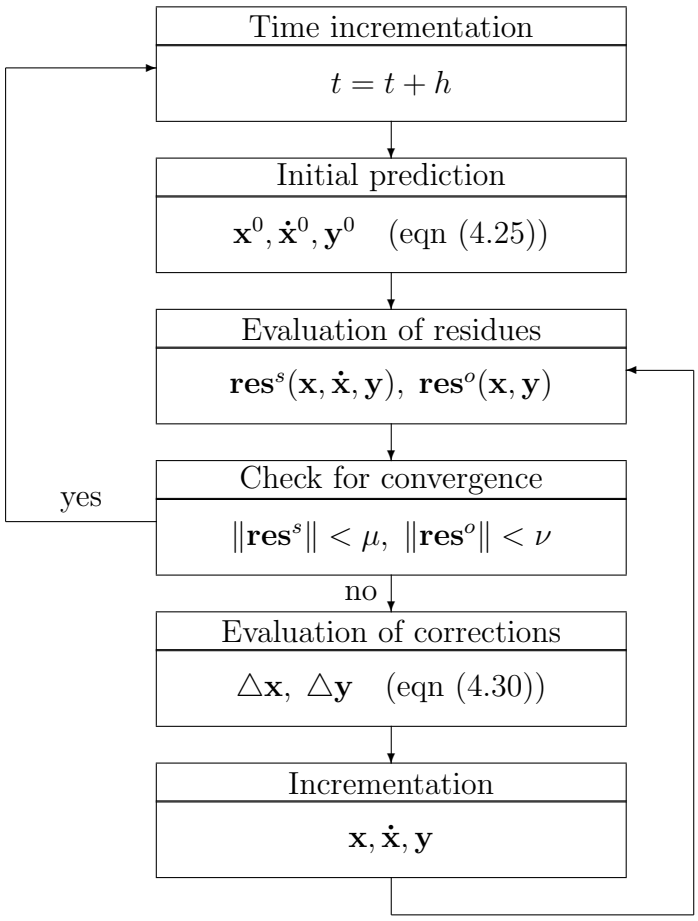


Figure 4.5: Integration scheme for a control system.

and introducing this result in the first equation, we obtain the ODE system:

$$\dot{\mathbf{x}} + \mathbf{A}\,\mathbf{x} = \mathbf{0} \qquad \text{with} \qquad \mathbf{A} = -\mathbf{f}_{\mathbf{x}}^{s} - \mathbf{f}_{\mathbf{u}}^{s}\,\mathbf{L}^{io}\,\left(\mathbf{I} - \mathbf{f}_{\mathbf{u}}^{o}\,\mathbf{L}^{io}\right)^{-1}\,\mathbf{f}_{\mathbf{x}}^{o} \tag{4.33}$$

Independent modal equations can be derived by a further linear transformation. Assuming that the state matrix $\mathbf{A}$ is diagonalized by the matrix of eigenvectors $\mathbf{T}_{\mathbf{x}}$, we define the modal coordinates $\mathbf{x}^*$ and $\mathbf{y}^*$:

$$\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{T}_{\mathbf{x}} & \mathbf{0} \\ -\mathbf{C}\,\mathbf{T}_{\mathbf{x}} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x}^* \\ \mathbf{y}^* \end{bmatrix} \tag{4.34}$$

so that (4.31) is equivalent to:

$$\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{x}}^* \\ \dot{\mathbf{y}}^* \end{bmatrix} + \begin{bmatrix} \mathbf{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x}^* \\ \mathbf{y}^* \end{bmatrix} = \mathbf{0} \tag{4.35}$$

where $\mathbf{\Sigma}$ is the diagonal matrix of the eigenvalues of $\mathbf{A}$. It is sufficient to study the stability of the algorithm for each independent scalar equation.

We begin with the *equation of a dynamic state*:

$$\dot{x}^* + \sigma\,x^* = 0 \tag{4.36}$$

which is discretized according to the modified residual equation (4.20)

$$(1-\delta_m)\,\dot{x}^*_{n+1} + \delta_m\,\dot{x}^*_n + (1-\delta_f)\,\sigma\,x^*_{n+1} + \delta_f\,\sigma\,x^*_n = 0 \tag{4.37}$$

Combining this equation with the integration formula (4.19), we get the matrix relation

$$\begin{bmatrix} (1-\delta_f)\,\sigma & (1-\delta_m) \\ 1 & -\theta h \end{bmatrix} \begin{bmatrix} x^* \\ \dot{x}^* \end{bmatrix}_{n+1} + \begin{bmatrix} \delta_f\,\sigma & \delta_m \\ -1 & -(1-\theta)h \end{bmatrix} \begin{bmatrix} x^* \\ \dot{x}^* \end{bmatrix}_{n} = 0 \tag{4.38}$$

and we seek its eigensolutions, characterized by a synchronous behavior:

$$\begin{bmatrix} x^* \\ \dot{x}^* \end{bmatrix}_n = \varphi_n \begin{bmatrix} x^*_0 \\ \dot{x}^*_0 \end{bmatrix} \qquad \text{with} \qquad \varphi_{n+1} = \zeta\,\varphi_n \tag{4.39}$$

φ is the amplitude, $[x^*_0 \;\; \dot{x}^*_0]^T$ is the eigenvector, and ζ is the eigenvalue. The characteristic equation is

$$P^{\mathbf{x}}_{\sigma h} = \det \begin{bmatrix} P_{\delta_f}(\zeta)\sigma & P_{\delta_m}(\zeta) \\ P_1(\zeta) & -P_\theta(\zeta)h \end{bmatrix} = 0 \tag{4.40}$$

with the polynomials:

$$\begin{aligned} P_\epsilon(\zeta) &= (1-\epsilon)\,\zeta + \epsilon \qquad \epsilon = \{\delta_m, \delta_f\} \\ P_\theta(\zeta) &= \theta\,\zeta + (1-\theta) \\ P_1(\zeta) &= \zeta - 1 \end{aligned} \tag{4.41}$$

Developing the determinant, we obtain:

$$P^{\mathbf{x}}_{\sigma h} = P_{\delta_f}\; P_\theta\; \sigma h + P_1\, P_{\delta_m} = 0 \tag{4.42}$$

The absolute stability is characterized by the spectral radius when $\sigma h \to \infty$:

$$P^{\mathbf{x}}_{\infty} = P_{\delta_f}\; P_\theta = 0 \tag{4.43}$$

This equation has two simple roots:

$$\zeta_1^{\mathbf{x}} = \frac{-\delta_f}{1-\delta_f} \qquad \zeta_2^{\mathbf{x}} = -\frac{1-\theta}{\theta} \tag{4.44}$$

where $\zeta_1^{\mathbf{x}}$ is the spurious root (equivalent to the spurious root $\zeta_1^{\mathbf{q}}$ associated with the simulation of a mechanical system), and $\zeta_2^{\mathbf{x}}$ is the principal root. With the optimal parameters for second-order accuracy $\theta = 1/2 + \delta_{fm}$, we get

$$\zeta_2^{\mathbf{x}} = -\frac{1-2\,\delta_{fm}}{1+2\,\delta_{fm}} \tag{4.45}$$

Finally, in the system (4.35), the *algebraic equation* associated with an output

$$y^* = 0 \tag{4.46}$$

is discretized according to equation (4.21)

$$(1-\delta_f)\, y^*_{n+1} + \delta_f\, y^*_n = 0 \tag{4.47}$$

The single eigenvalue of the amplification matrix is the spurious root $\zeta_1^{\mathbf{x}}$.

As a conclusion, $|\zeta_1^{\mathbf{x}}| < 1$ requires $\delta_f < 1/2$, and $|\zeta_2^{\mathbf{x}}| < 1$ requires $\delta_{fm} > 0$, so that the stability is guaranteed for:

$$\delta_m < \delta_f < \frac{1}{2} \tag{4.48}$$

Equivalent results were obtained by Jansen *et al.* [JWH00]. It is easily demonstrated that the optimal condition $|\zeta_1^{\mathbf{x}}| = |\zeta_2^{\mathbf{x}}| = \rho_\infty^{\mathbf{x}}$ leads to

$$\delta_m = \frac{1}{2}\left(\frac{3\,\rho_\infty^{\mathbf{x}} - 1}{\rho_\infty^{\mathbf{x}} + 1}\right) \qquad \text{and} \qquad \delta_f = \frac{\rho_\infty^{\mathbf{x}}}{\rho_\infty^{\mathbf{x}} + 1} \tag{4.49}$$

These formulae are different from the optimal formulae (3.30) obtained for the simulation of a purely mechanical system.

4.2 Integrated simulation of mechatronic systems

The formalisms developed for the analysis of flexible mechanisms (chapter 3) and for the analysis of control systems (previous section) can be combined for the analysis of mechatronic systems.

4.2.1 Coupled equations of motion

In a mechatronic system, the input/output variables of the control system are directly connected with the mechanical dofs and their derivatives. The *sensor measurements* $\mathbf{w}^m$ are associated with the displacements, velocities and accelerations:

$$\mathbf{w}^m = \mathbf{L}^{m\mathbf{q}}\,\mathbf{q} + \mathbf{L}^{m\dot{\mathbf{q}}}\,\dot{\mathbf{q}} + \mathbf{L}^{m\ddot{\mathbf{q}}}\,\ddot{\mathbf{q}} \tag{4.50}$$

where $\mathbf{L}^{m\mathbf{q}}$, $\mathbf{L}^{m\dot{\mathbf{q}}}$, $\mathbf{L}^{m\ddot{\mathbf{q}}}$ are sensor localization matrices. The expression of the input variables follow from (4.8)

$$\mathbf{u} = \mathbf{L}^{i\mathbf{q}}\,\mathbf{q} + \mathbf{L}^{i\dot{\mathbf{q}}}\,\dot{\mathbf{q}} + \mathbf{L}^{i\ddot{\mathbf{q}}}\,\ddot{\mathbf{q}} + \mathbf{L}^{io}\,\mathbf{y} \tag{4.51}$$

with $\mathbf{L}^{i\mathbf{q}} = \mathbf{L}^{im}\,\mathbf{L}^{m\mathbf{q}}$, $\mathbf{L}^{i\dot{\mathbf{q}}} = \mathbf{L}^{im}\,\mathbf{L}^{m\dot{\mathbf{q}}}$, and $\mathbf{L}^{i\ddot{\mathbf{q}}} = \mathbf{L}^{im}\,\mathbf{L}^{m\ddot{\mathbf{q}}}$. The *generalized forces of the actuators* $\mathbf{g}^a$ produce the mechanical virtual work

$$\delta W^a = (\mathbf{L}^{\mathbf{q}a}\,\mathbf{g}^a)^T\,\delta\mathbf{q} = (\mathbf{L}^{\mathbf{q}a}\,\mathbf{L}^{ao}\,\mathbf{y})^T\,\delta\mathbf{q} = (\mathbf{L}^{\mathbf{q}o}\,\mathbf{y})^T\,\delta\mathbf{q} \tag{4.52}$$

where $\mathbf{L}^{\mathbf{q}a}$ and $\mathbf{L}^{\mathbf{q}o} = \mathbf{L}^{\mathbf{q}a}\,\mathbf{L}^{ao}$ are the actuator and output localization matrices.

The whole set of coupled equations is then:

$$\mathbf{M}\,\ddot{\mathbf{q}} + \mathbf{\Phi}_{\mathbf{q}}^T\,(k\,\boldsymbol{\lambda} + p\,\mathbf{\Phi}) - \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}, t) - \mathbf{L}^{\mathbf{q}o}\,\mathbf{y} = \mathbf{0} \tag{4.53}$$

$$k\,\mathbf{\Phi}(\mathbf{q}) = \mathbf{0} \tag{4.54}$$

$$\dot{\mathbf{x}} - \mathbf{f}^s(\mathbf{u}, \mathbf{x}, t) = \mathbf{0} \tag{4.55}$$

$$\mathbf{y} - \mathbf{f}^o(\mathbf{u}, \mathbf{x}, t) = \mathbf{0} \tag{4.56}$$

$$\mathbf{u} - \mathbf{L}^{i\mathbf{q}}\,\mathbf{q} - \mathbf{L}^{i\dot{\mathbf{q}}}\,\dot{\mathbf{q}} - \mathbf{L}^{i\ddot{\mathbf{q}}}\,\ddot{\mathbf{q}} - \mathbf{L}^{io}\,\mathbf{y} = \mathbf{0} \tag{4.57}$$

Equation (4.53) represents the dynamics of the mechanical system, equation (4.54), the kinematic constraints, equation (4.55), the state dynamics, equations (4.56) the algebraic output equations, and (4.57) the input localizations.

4.2.2 Time-integration algorithm

A usual method to simulate such a coupled system of equations is to reformulate the second-order DAEs of the mechanism as first-order DAEs, by the introduction of the mechanical state vector $\mathbf{x}^{mech} = \left[\mathbf{q}^T \ \dot{\mathbf{q}}^T\right]^T$. Any DAE integration scheme can then be exploited, such as multistep or Runge-Kutta methods [BCP96], provided that the algebraic content of the coupled equations leads to a "reasonable" index. This algebraic content comes from the kinematic constraints (4.54), and from the input/output equations (4.56),(4.57). Considering an *unconstrained mechanical system*, under a few technical assumptions, the coupled system is equivalent to a semi-explicit index-1 system, so that the simulation can be done with the same convergence and stability properties as if the integrator were applied to stiff ODEs. In the presence of *kinematic constraints* formulated at the position level, the system has an index-3 structure, and standard integration methods may lead to unreliable results [BCP96]. Therefore, the index of the system should be reduced, either using a constraint elimination technique, or a constraint differentiation technique. In the literature, numerous methods have been proposed for this difficult problem, which is still the object of intensive research.

We have seen that the generalized-α method, known in structural dynamics, can be considered for the simulation of *constrained flexible multibody systems*. Theoretical convergence and stability results are available, and the method appears to be quite efficient in practice. Moreover, the generalized-α method is also applicable for *state-space models*, as shown in section 4.1.

Hence, we shall demonstrate that the generalized-α method leads to a unified framework for the *simulation of mechatronic systems*. Many convergence and stability properties of the original scheme developed for mechanical systems are inherited thanks to this approach, as well as the systematic treatment of the kinematic constraints. The numerical algorithm is presented and analyzed in the following.

Combining the discretized mechanical equations (section 3.3) and the discretized state equations (section 4.1), we obtain the following coupled set of equa-

tions:

$$\left\{\begin{array}{lll} (1-\alpha_m)\,(\mathbf{M}\,\ddot{\mathbf{q}})_{n+1} + \alpha_m\,(\mathbf{M}\,\ddot{\mathbf{q}})_n \;+\; (1-\alpha_f)\,\mathbf{g}^*_{n+1} + \alpha_f\,\mathbf{g}^*_n & = \mathbf{0} \\ \qquad\qquad\qquad\qquad\qquad\qquad (1-\alpha_f)\,k\,\boldsymbol{\Phi}_{n+1} + \alpha_f\,k\,\boldsymbol{\Phi}_n & = \mathbf{0} \\ \mathbf{q}_{n+1} = \mathbf{q}_n + h\,\dot{\mathbf{q}}_n + h^2\,(\frac{1}{2}-\beta)\,\ddot{\mathbf{q}}_n + h^2\,\beta\,\ddot{\mathbf{q}}_{n+1} & \\ \dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + h\,(1-\gamma)\,\ddot{\mathbf{q}}_n + h\,\gamma\,\ddot{\mathbf{q}}_{n+1} & \end{array}\right.$$

$$\left\{\begin{array}{lll} (1-\delta_m)\,\dot{\mathbf{x}}_{n+1} + \delta_m\,\dot{\mathbf{x}}_n - \;(1-\delta_f)\,\mathbf{f}^s_{n+1} - \delta_f\,\mathbf{f}^s_n & = \mathbf{0} \\ \qquad\qquad\qquad\qquad\qquad\qquad (1-\delta_f)\,(\mathbf{y}_{n+1} - \mathbf{f}^o_{n+1}) + \delta_f\,(\mathbf{y}_n - \mathbf{f}^o_n) & = \mathbf{0} \\ \mathbf{x}_{n+1} = \mathbf{x}_n + h\,(1-\theta)\,\dot{\mathbf{x}}_n + h\,\theta\,\dot{\mathbf{x}}_{n+1} & \end{array}\right. \tag{4.58}$$

with the discretized inputs:

$$\mathbf{u}_{n+1} = \mathbf{L}^{i\mathbf{q}}\,\mathbf{q}_{n+1} + \mathbf{L}^{i\dot{\mathbf{q}}}\,\dot{\mathbf{q}}_{n+1} + \mathbf{L}^{i\ddot{\mathbf{q}}}\,\ddot{\mathbf{q}}_{n+1} + \mathbf{L}^{io}\,\mathbf{y}_{n+1} \tag{4.59}$$

where $\mathbf{g}^*$ is a notation for $\boldsymbol{\Phi}^T_{\mathbf{q}}\,(k\,\boldsymbol{\lambda} + p\,\boldsymbol{\Phi}) - \mathbf{g}(\mathbf{q},\dot{\mathbf{q}}) - \mathbf{L}^{\mathbf{q}o}\,\mathbf{y}$. The first set of equations contains the discretized dynamic equations associated with the mechanical dofs and the Newmark formulae. The second contains the discretized state equations and their time-integration formula. At this level, two remarks may be formulated, which suggest a reformulation of those equations.

Remark 4.1 *In order to uniformize the treatment of the state and displacement variables, the dummy dynamic variables* $\mathbf{z}$ *are introduced:*

$$\mathbf{z}(t) = \int_0^t \mathbf{x}(\tau)\,d\tau \tag{4.60}$$

The value of $\mathbf{z}$ *is only meaningful at the velocity level* ($\dot{\mathbf{z}} = \mathbf{x}$) *and at the acceleration level* ($\ddot{\mathbf{z}} = \dot{\mathbf{x}}$)*; nevertheless, an artificial displacement-like Newmark formula is added for* $\mathbf{z}$*. Moreover, we impose a constraint on the choice of the algorithmic parameters*

$$\delta_m = \alpha_m, \qquad \delta_f = \alpha_f, \qquad \theta = \gamma \tag{4.61}$$

Remark 4.2 *According to Remark 3.1,* $\ddot{\mathbf{q}}_{n+1}$ *is a poor approximation for* $\ddot{\mathbf{q}}(t_{n+1})$*. Therefore, we exclude the acceleration term from the input equation,*

$$\mathbf{u} = \mathbf{L}^{i\mathbf{q}}\,\mathbf{q} + \mathbf{L}^{i\dot{\mathbf{q}}}\,\dot{\mathbf{q}} + \mathbf{L}^{io}\,\mathbf{y} \tag{4.62}$$

and we define an observer equation for each acceleration measurement $\ddot{q}^i$*:*

$$y^a = \ddot{q}^i \tag{4.63}$$

This equation is discretized according to

$$(1-\alpha_f)\; y^a_{n+1} + \alpha_f\; y^a_n = (1-\alpha_m)\; \ddot{q}^i_{n+1} + \alpha_m\; \ddot{q}^i_n \tag{4.64}$$

so that y^a_{n+1} is an order 2 approximation for $\ddot{q}^i(t_{n+1})$, which can be safely connected to any input port. The global output equation becomes

$$\mathbf{y} = \mathbf{f}^o(\mathbf{u}, \mathbf{x}, t) + \mathbf{L}^{o\ddot{\mathbf{q}}}\; \ddot{\mathbf{q}} \tag{4.65}$$

The discretized equations of motion are then restated,

$$\left\{\begin{array}{llllll}
(1-\alpha_m)\;(\mathbf{M}\;\ddot{\mathbf{q}})_{n+1} & + & \alpha_m\;(\mathbf{M}\;\ddot{\mathbf{q}})_n & + & (1-\alpha_f)\;\mathbf{g}^*_{n+1} + \alpha_f\;\mathbf{g}^*_n & = \mathbf{0} \\
 & & & & (1-\alpha_f)\;k\;\mathbf{\Phi}_{n+1} + \alpha_f\;k\;\mathbf{\Phi}_n & = \mathbf{0} \\
(1-\alpha_m)\;\ddot{\mathbf{z}}_{n+1} & + & \alpha_m\;\ddot{\mathbf{z}}_n & - & (1-\alpha_f)\;\mathbf{f}^s_{n+1} - \alpha_f\;\mathbf{f}^s_n & = \mathbf{0} \\
-(1-\alpha_m)\;\mathbf{L}^{o\ddot{\mathbf{q}}}\;\ddot{\mathbf{q}}_{n+1} & - & \alpha_m\;\mathbf{L}^{o\ddot{\mathbf{q}}}\;\ddot{\mathbf{q}}_n & & & \\
 & & & & + (1-\alpha_f)\;(\mathbf{y}_{n+1} - \mathbf{f}^o_{n+1}) + \alpha_f\;(\mathbf{y}_n - \mathbf{f}^o_n) & = \mathbf{0}
\end{array}\right.$$

$$\left\{\begin{array}{l}
\mathbf{q}_{n+1} = \mathbf{q}_n + h\;\dot{\mathbf{q}}_n + h^2\;(\frac{1}{2}-\beta)\;\ddot{\mathbf{q}}_n + h^2\;\beta\;\ddot{\mathbf{q}}_{n+1} \\
\mathbf{z}_{n+1} = \mathbf{z}_n + h\;\dot{\mathbf{z}}_n + h^2\;(\frac{1}{2}-\beta)\;\ddot{\mathbf{z}}_n + h^2\;\beta\;\ddot{\mathbf{z}}_{n+1} \\
\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + h\;(1-\gamma)\;\ddot{\mathbf{q}}_n + h\;\gamma\;\ddot{\mathbf{q}}_{n+1} \\
\dot{\mathbf{z}}_{n+1} = \dot{\mathbf{z}}_n + h\;(1-\gamma)\;\ddot{\mathbf{z}}_n + h\;\gamma\;\ddot{\mathbf{z}}_{n+1}
\end{array}\right. \tag{4.66}$$

The advantage of this equivalent expression comes from the observation that $\mathbf{q}$ and $\mathbf{z}$ receive exactly the same treatment. Hence, the formulation, the implementation and the theoretical analysis of the algorithm are highly simplified.

The prediction for $\mathbf{q}_{n+1}, \mathbf{z}_{n+1}, \dot{\mathbf{q}}_{n+1}, \dot{\mathbf{z}}_{n+1}$ follows from the standard Newmark formulae with the zero acceleration assumption $\ddot{\mathbf{q}}^0_{n+1} = \mathbf{0}$, $\ddot{\mathbf{z}}^0_{n+1} = \mathbf{0}$. Using the linearized form of the input equation (4.62):

$$\Delta\mathbf{u} = \left(\mathbf{L}^{i\mathbf{q}} + \frac{\gamma}{\beta h}\;\mathbf{L}^{i\dot{\mathbf{q}}}\right)\;\triangle\mathbf{q}\; + \mathbf{L}^{io}\;\Delta\mathbf{y} \tag{4.67}$$

the correction equation follows:

$$\underline{\mathbf{S}_t}\;\left[\begin{array}{c}\triangle\mathbf{q}\\ \triangle\boldsymbol{\lambda}\\ \triangle\mathbf{z}\\ \triangle\mathbf{y}\end{array}\right] = \left[\begin{array}{c}-\mathbf{res}^{\mathbf{q}}_k\\ -\mathbf{res}^{\mathbf{\Phi}}_k\\ -\mathbf{res}^{s}_k\\ -\mathbf{res}^{o}_k\end{array}\right] \tag{4.68}$$

where

$$\underline{\mathbf{S}_t} = (1-\alpha_f)\,\underline{\mathbf{K}_t} + (1-\alpha_f)\,\frac{\gamma}{\beta h}\,\underline{\mathbf{C}_t} + (1-\alpha_m)\,\frac{1}{\beta h^2}\,\underline{\mathbf{M}_t} \tag{4.69}$$

$\underline{\mathbf{K}_t}$, $\underline{\mathbf{C}_t}$, $\underline{\mathbf{M}_t}$ are respectively given by

$$\left[\begin{array}{cc|cc} \mathbf{K}_t & k\,\mathbf{\Phi}_\mathbf{q}^T & \mathbf{0} & -\mathbf{L}^{\mathbf{q}o} \\ k\,\mathbf{\Phi}_\mathbf{q} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline -\mathbf{f}_\mathbf{u}^s\,\mathbf{L}^{i\mathbf{q}} & \mathbf{0} & \mathbf{0} & -\mathbf{f}_\mathbf{u}^s\,\mathbf{L}^{io} \\ -\mathbf{f}_\mathbf{u}^o\,\mathbf{L}^{i\mathbf{q}} & \mathbf{0} & \mathbf{0} & \mathbf{I}-\mathbf{f}_\mathbf{u}^o\,\mathbf{L}^{io} \end{array}\right], \left[\begin{array}{cc|cc} \mathbf{C}_t & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline -\mathbf{f}_\mathbf{u}^s\,\mathbf{L}^{i\dot{\mathbf{q}}} & \mathbf{0} & -\mathbf{f}_\mathbf{x}^s & \mathbf{0} \\ -\mathbf{f}_\mathbf{u}^o\,\mathbf{L}^{i\dot{\mathbf{q}}} & \mathbf{0} & -\mathbf{f}_\mathbf{x}^o & \mathbf{0} \end{array}\right], \left[\begin{array}{cc|cc} \mathbf{M} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ -\mathbf{L}^{o\ddot{\mathbf{q}}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right] \tag{4.70}$$

The correction equation involves non-symmetric matrices. Thanks to the penalization term, $\mathbf{K}_t$ is positive definite and it is possible to apply a direct solver without pivoting strategy. Good performances were observed with a non-symmetric direct solver optimized for sparse matrices. This choice is reasonable since a mechatronic system usually involves no more than a few hundreds dofs. For systems with much more dofs, iterative solvers could be advantageously applied.

The time-integration algorithm is described in Figure 4.6. *In order to analyze its stability and convergence properties*, it is equivalent to analyze the dynamic system after elimination of the input and output variables from the state equations. Under Assumption 4.1, the input and output variables defined by (4.62) and (4.65) can be formulated in explicit format (4.14), (4.15), and replaced in the dynamic equations (4.53), (4.55):

$$\mathbf{M}\,\ddot{\mathbf{q}} - \mathbf{g}(\mathbf{q},\dot{\mathbf{q}},t) + \mathbf{\Phi}_\mathbf{q}^T\,(k\,\boldsymbol{\lambda} + p\,\mathbf{\Phi}) - \mathbf{L}^{\mathbf{q}o}\,\widetilde{\mathbf{f}}^o(\mathbf{x},\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}},t) = \mathbf{0} \tag{4.71}$$

$$k\,\mathbf{\Phi}(\mathbf{q}) = \mathbf{0} \tag{4.72}$$

$$\dot{\mathbf{x}} - \widetilde{\mathbf{f}}^s(\mathbf{x},\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}},t) = \mathbf{0} \tag{4.73}$$

where $\widetilde{\mathbf{f}}^s(\mathbf{x},\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}},t) = \mathbf{f}^s(\widetilde{\mathbf{f}^\mathbf{u}}(\mathbf{x},\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}},t),\mathbf{x},t)$.

4.2.3 Linear stability analysis: the unconstrained case

For a purely mechanical system, the linear stability of the generalized-α method has been analyzed under the assumption of a negligible damping matrix. In case of a mechatronic system, the dynamics of the state variables is described by first-order differential equations, and this assumption is no more relevant. For this reason, an extended linear stability analysis is necessary, which is realized here for the unconstrained case, and in the next section for the constrained case.

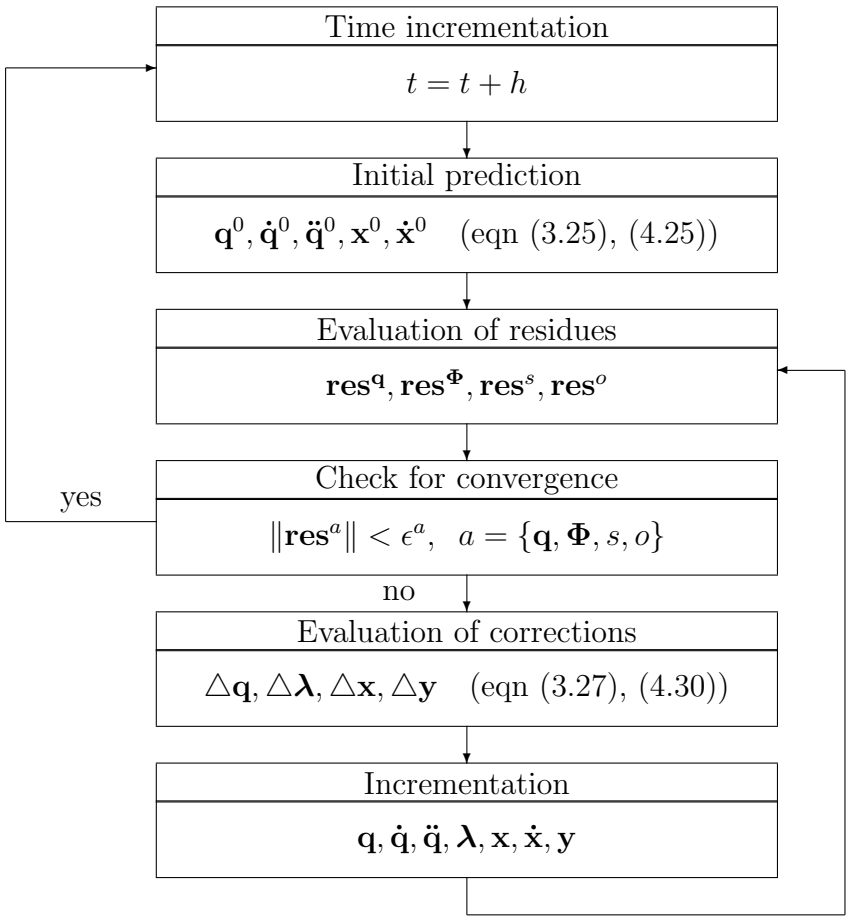


Figure 4.6: Time-integration algorithm for mechatronic systems

In order to investigate the stability of the algorithm when the mechanical system is unconstrained, the linearized form of equations (4.71) and (4.73) is considered:

$$\begin{aligned} \widetilde{\mathbf{M}}\,\ddot{\mathbf{q}} + \mathbf{C}\,\dot{\mathbf{q}} + \mathbf{K}\,\mathbf{q} + \mathbf{G}\,\mathbf{x} &= \mathbf{0} \\ \dot{\mathbf{x}} + \mathbf{A}\,\mathbf{x} + \mathbf{P}\,\mathbf{q} + \mathbf{D}\,\dot{\mathbf{q}} + \mathbf{F}\,\ddot{\mathbf{q}} &= \mathbf{0} \end{aligned} \tag{4.74}$$

The matrices $\widetilde{\mathbf{M}}$, $\mathbf{C}$ and $\mathbf{K}$ include the possible contributions of the direct feedback control actions $\widetilde{\mathbf{f}}^{o}_{\ddot{\mathbf{q}}}$, $\widetilde{\mathbf{f}}^{o}_{\dot{\mathbf{q}}}$ and $\widetilde{\mathbf{f}}^{o}_{\mathbf{q}}$, respectively (*e.g.* $\widetilde{\mathbf{M}} \neq \mathbf{M}$). Hence, $\widetilde{\mathbf{M}}$, $\mathbf{C}$ and $\mathbf{K}$ are not necessarily symmetric positive definite, which is an important difference compared to the mass, damping, and stiffness matrices of a passive mechanical system.

The velocity term $\mathbf{C}\,\dot{\mathbf{q}}$ may come from the internal damping of the material, or from a direct feedback action. It is expected that this latter contribution has a stabilizing effect, otherwise, the whole dynamic system may get unstable as well as the numerical simulation. Therefore, the case $\mathbf{C} = \mathbf{0}$ can be seen as the worst case situation, where stability in the simulation results is desirable; the damping matrix $\mathbf{C}$ is thus omitted in this analysis.

The difficulty may be reduced by diagonalization of matrices $\left(\widetilde{\mathbf{M}}^{-1}\,\mathbf{K}\right)$ and $\mathbf{A}$:

$$\mathbf{\Omega}^2 = \mathbf{T}_{\mathbf{q}}^{-1}\,\widetilde{\mathbf{M}}^{-1}\,\mathbf{K}\,\mathbf{T}_{\mathbf{q}} \qquad \text{and} \qquad \mathbf{\Sigma} = \mathbf{T}_{\mathbf{x}}^{-1}\,\mathbf{A}\,\mathbf{T}_{\mathbf{x}} \tag{4.75}$$

where $\mathbf{\Omega}$ and $\mathbf{\Sigma}$ are diagonal matrices, $\mathbf{T}_{\mathbf{q}}$ and $\mathbf{T}_{\mathbf{x}}$ are transformation matrices. Defining the modal coordinates $\mathbf{q}^*$ and $\mathbf{x}^*$:

$$\mathbf{q} = \mathbf{T}_{\mathbf{q}}\,\mathbf{q}^* \qquad \text{and} \qquad \mathbf{x} = \mathbf{T}_{\mathbf{x}}\,\mathbf{x}^* \tag{4.76}$$

and the coupling matrices:

$$\mathbf{\Gamma} = \mathbf{T}_{\mathbf{q}}^{-1}\,\mathbf{G}\,\mathbf{T}_{\mathbf{x}}, \quad \mathbf{\Pi} = \mathbf{T}_{\mathbf{x}}^{-1}\,\mathbf{P}\,\mathbf{T}_{\mathbf{q}}, \quad \mathbf{\Delta} = \mathbf{T}_{\mathbf{x}}^{-1}\,\mathbf{D}\,\mathbf{T}_{\mathbf{q}}, \quad \mathbf{\Upsilon} = \mathbf{T}_{\mathbf{x}}^{-1}\,\mathbf{F}\,\mathbf{T}_{\mathbf{q}} \tag{4.77}$$

we obtain the equivalent system:

$$\begin{aligned} \ddot{\mathbf{q}}^* + \mathbf{\Omega}^2\,\mathbf{q}^* + \mathbf{\Gamma}\,\mathbf{x}^* &= \mathbf{0} \\ \dot{\mathbf{x}}^* + \mathbf{\Sigma}\,\mathbf{x}^* + \mathbf{\Pi}\,\mathbf{q}^* + \mathbf{\Delta}\,\dot{\mathbf{q}}^* + \mathbf{\Upsilon}\,\ddot{\mathbf{q}}^* &= \mathbf{0} \end{aligned} \tag{4.78}$$

where the coupling matrices $\mathbf{\Gamma}$, $\mathbf{\Pi}$, $\mathbf{\Delta}$ and $\mathbf{\Upsilon}$ have no specific structure, that could be exploited for a component-wise analysis.

Those equations are discretized according to (4.58):

$$\left[\begin{array}{ccc|cc} (1-\alpha_f)\,\mathbf{\Omega}^2 & \mathbf{0} & (1-\alpha_m)\,\mathbf{I} & (1-\alpha_f)\,\mathbf{\Gamma} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & -\beta h^2\,\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & -\gamma h\,\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \hline (1-\alpha_f)\,\mathbf{\Pi} & (1-\alpha_f)\,\mathbf{\Delta} & (1-\alpha_m)\,\mathbf{\Upsilon} & (1-\alpha_f)\,\mathbf{\Sigma} & (1-\alpha_m)\,\mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & -\gamma h\,\mathbf{I} \end{array}\right] \left[\begin{array}{c} \mathbf{q}^* \\ \dot{\mathbf{q}}^* \\ \ddot{\mathbf{q}}^* \\ \hline \mathbf{x}^* \\ \dot{\mathbf{x}}^* \end{array}\right]_{n+1}$$

$$+ \left[\begin{array}{ccc|cc} \alpha_f\,\mathbf{\Omega}^2 & \mathbf{0} & \alpha_m\,\mathbf{I} & \alpha_f\,\mathbf{\Gamma} & \mathbf{0} \\ -\mathbf{I} & -h\,\mathbf{I} & -(\tfrac{1}{2}-\beta)h^2\,\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{I} & -(1-\gamma)h\,\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \hline \alpha_f\,\mathbf{\Pi} & \alpha_f\,\mathbf{\Delta} & \alpha_m\,\mathbf{\Upsilon} & \alpha_f\,\mathbf{\Sigma} & \alpha_m\,\mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & -(1-\gamma)h\,\mathbf{I} \end{array}\right] \left[\begin{array}{c} \mathbf{q}^* \\ \dot{\mathbf{q}}^* \\ \ddot{\mathbf{q}}^* \\ \hline \mathbf{x}^* \\ \dot{\mathbf{x}}^* \end{array}\right]_n = \mathbf{0} \tag{4.79}$$

where $\mathbf{I}$ and $\mathbf{0}$ still denote identity and null matrices of appropriate dimensions.

We seek the eigensolutions of this difference equation, characterized by a synchronous behavior:

$$\left[\begin{array}{c} \mathbf{q}^* \\ \dot{\mathbf{q}}^* \\ \ddot{\mathbf{q}}^* \\ \mathbf{x}^* \\ \dot{\mathbf{x}}^* \end{array}\right]_n = \varphi_n \left[\begin{array}{c} \mathbf{q}_0^* \\ \dot{\mathbf{q}}_0^* \\ \ddot{\mathbf{q}}_0^* \\ \mathbf{x}_0^* \\ \dot{\mathbf{x}}_0^* \end{array}\right] \qquad \text{with} \qquad \varphi_{n+1} = \zeta\,\varphi_n \tag{4.80}$$

so that the characteristic equation is

$$\det \left[\begin{array}{ccc|cc} P_{\alpha_f}(\zeta)\,\mathbf{\Omega}^2 & \mathbf{0} & P_{\alpha_m}(\zeta)\,\mathbf{I} & P_{\alpha_f}(\zeta)\,\mathbf{\Gamma} & \mathbf{0} \\ P_1(\zeta)\mathbf{I} & -h\,\mathbf{I} & -P_\beta(\zeta)h^2\,\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & P_1(\zeta)\mathbf{I} & -P_\gamma(\zeta)h\,\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \hline P_{\alpha_f}(\zeta)\,\mathbf{\Pi} & P_{\alpha_f}(\zeta)\,\mathbf{\Delta} & P_{\alpha_m}(\zeta)\,\mathbf{\Upsilon} & P_{\alpha_f}(\zeta)\,\mathbf{\Sigma} & P_{\alpha_m}(\zeta)\,\mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & P_1(\zeta)\mathbf{I} & -P_\gamma(\zeta)h\,\mathbf{I} \end{array}\right] = 0 \tag{4.81}$$

with the polynomials $P_{\alpha_m}, P_{\alpha_f}, P_\beta, P_\gamma, P_1$ defined in equation (3.41).

If $\mathbf{\Gamma} = \mathbf{0}$ or $\mathbf{\Pi} = \mathbf{\Delta} = \mathbf{\Upsilon} = \mathbf{0}$, a block triangular structure appears, and the characteristic equation may be splitted into one equation for each block of the diagonal, indicating an uncoupling in the time-integration of both subsystems. In this case, the stability of the global integration scheme follows from the stability

of the scheme applied to the independent subsystems, which has been analyzed before.

In any other situation, the analysis cannot be pursued without further assumption. As proposed by Gear and Wells in a similar context [GW84], let us consider the scalar case (one mechanical dof q and one state variable x). Then, the characteristic equation becomes:

$$\det \left[\begin{array}{ccc|cc} P_{\alpha_f}(\zeta)\,\omega^2 & 0 & P_{\alpha_m}(\zeta) & P_{\alpha_f}(\zeta)\,g & 0 \\ P_1(\zeta) & -h & -P_\beta(\zeta)h^2 & 0 & 0 \\ 0 & P_1(\zeta) & -P_\gamma(\zeta)h & 0 & 0 \\ \hline P_{\alpha_f}(\zeta)\,p & P_{\alpha_f}(\zeta)\,d & P_{\alpha_m}(\zeta)\,f & P_{\alpha_f}(\zeta)\,\sigma & P_{\alpha_m}(\zeta) \\ 0 & 0 & 0 & P_1(\zeta) & -P_\gamma(\zeta)h \end{array}\right] = 0 \qquad (4.82)$$

In order to analyze the stability in the presence of stiff dynamics in the mechanical systems and/or in the control system, the characteristic equation is developed in three subcases:

1. $\omega h \to \infty$:

$$P_{\alpha_f}(\zeta)\,\omega^2 \det \left[\begin{array}{cc|cc} -h & -P_\beta(\zeta)h^2 & 0 & 0 \\ P_1(\zeta) & -P_\gamma(\zeta)h & 0 & 0 \\ \hline P_{\alpha_f}(\zeta)\,d & P_{\alpha_m}(\zeta)\,f & P_{\alpha_f}(\zeta)\,\sigma & P_{\alpha_m}(\zeta) \\ 0 & 0 & P_1(\zeta) & -P_\gamma(\zeta)h \end{array}\right] = 0 \qquad (4.83)$$

   which is equivalent to $P^{\mathbf{q}}_{\infty}(\zeta)\; P^{\mathbf{x}}_{\sigma h}(\zeta) = 0$.

2. $\sigma h \to \infty$:

$$P_{\alpha_f}(\zeta)\,\sigma \det \left[\begin{array}{ccc|c} P_{\alpha_f}(\zeta)\,\omega^2 & 0 & P_{\alpha_m}(\zeta) & 0 \\ P_1(\zeta) & -h & -P_\beta(\zeta)h^2 & 0 \\ 0 & P_1(\zeta) & -P_\gamma(\zeta)h & 0 \\ \hline 0 & 0 & 0 & -P_\gamma(\zeta)h \end{array}\right] = 0 \qquad (4.84)$$

   which is equivalent to $P^{\mathbf{q}}_{\omega h}(\zeta)\; P^{\mathbf{x}}_{\infty}(\zeta) = 0$.

3. $\omega h \to \infty$ and $\sigma h \to \infty$:

$$\begin{aligned} & P_{\alpha_f}(\zeta)\,\omega^2 \det \left[\begin{array}{cc|cc} -h & -P_\beta(\zeta)h^2 & 0 & 0 \\ P_1(\zeta) & -P_\gamma(\zeta)h & 0 & 0 \\ \hline P_{\alpha_f}(\zeta)\,d & P_{\alpha_m}(\zeta)\,f & P_{\alpha_f}(\zeta)\,\sigma & P_{\alpha_m}(\zeta) \\ 0 & 0 & P_1(\zeta) & -P_\gamma(\zeta)h \end{array}\right] \\ & +P_{\alpha_m}(\zeta) \det \left[\begin{array}{cc|cc} P_1(\zeta) & -h & 0 & 0 \\ 0 & P_1(\zeta) & 0 & 0 \\ \hline P_{\alpha_f}(\zeta)\,p & P_{\alpha_f}(\zeta)\,d & P_{\alpha_f}(\zeta)\,\sigma & P_{\alpha_m}(\zeta) \\ 0 & 0 & P_1(\zeta) & -P_\gamma(\zeta)h \end{array}\right] = 0 \end{aligned} \tag{4.85}$$

which is equivalent to $P^{\mathbf{q}}_{\omega h}(\zeta)\; P^{\mathbf{x}}_{\sigma h}(\zeta) = 0$.

As a conclusion, the linear stability of the integrator applied to the independent subsystems is a sufficient condition for the linear stability of the integrator applied to the unconstrained mechatronic system. We assume that this result, established for a scalar system (1 mechanical dof and 1 state), is a relevant indication for the stability of the algorithm in a more general context.

4.2.4 Linear stability analysis: the constrained case

The linearized equations of a constrained system are:

$$\begin{aligned} \widetilde{\mathbf{M}}\,\ddot{\mathbf{q}} + \mathbf{K}\,\mathbf{q} + \mathbf{B}^T\,\boldsymbol{\lambda} + \mathbf{G}\,\mathbf{x} &= \mathbf{0} \\ \mathbf{B}\,\mathbf{q} &= \mathbf{0} \\ \dot{\mathbf{x}} + \mathbf{A}\,\mathbf{x} + \mathbf{P}\,\mathbf{q} + \mathbf{D}\,\dot{\mathbf{q}} + \mathbf{F}\,\ddot{\mathbf{q}} &= \mathbf{0} \end{aligned} \tag{4.86}$$

As in the previous section, the damping term is neglected.

Inspired by section 3.3.3, it would be convenient to apply Theorem 3.1 page 52 in order to obtain a canonical form of the mechanical equations. This would be possible if the stiffness matrix $\mathbf{K}$ were real symmetric positive semi-definite, and the mass matrix $\widetilde{\mathbf{M}}$, real symmetric positive definite. Therefore, the developments of this section assume that $\mathbf{K}$ and $\widetilde{\mathbf{M}}$ satisfy those conditions, as in the passive case. This is always the case if no direct acceleration or displacement feedback is present.

Using the notations of sections 3.3.3 and 4.2.3, we define

$$\begin{aligned} &\boldsymbol{\Gamma}^r = \mathbf{T}^T_{\mathbf{q}r}\,\mathbf{G}\,\mathbf{T}_{\mathbf{x}}, \quad \boldsymbol{\Pi}^r = \mathbf{T}^{-1}_{\mathbf{x}}\,\mathbf{P}\,\mathbf{T}_{\mathbf{q}r}, \quad \boldsymbol{\Delta}^r = \mathbf{T}^{-1}_{\mathbf{x}}\,\mathbf{D}\,\mathbf{T}_{\mathbf{q}r} \quad \boldsymbol{\Upsilon}^r = \mathbf{T}^{-1}_{\mathbf{x}}\,\mathbf{F}\,\mathbf{T}_{\mathbf{q}r} \\ &\boldsymbol{\Gamma}^c = \mathbf{T}^T_{\mathbf{q}c}\,\mathbf{G}\,\mathbf{T}_{\mathbf{x}}, \quad \boldsymbol{\Pi}^c = \mathbf{T}^{-1}_{\mathbf{x}}\,\mathbf{P}\,\mathbf{T}_{\mathbf{q}c}, \quad \boldsymbol{\Delta}^c = \mathbf{T}^{-1}_{\mathbf{x}}\,\mathbf{D}\,\mathbf{T}_{\mathbf{q}c} \quad \boldsymbol{\Upsilon}^c = \mathbf{T}^{-1}_{\mathbf{x}}\,\mathbf{F}\,\mathbf{T}_{\mathbf{q}c} \end{aligned} \tag{4.87}$$

We obtain the following set of equations:

$$\begin{aligned} \ddot{\mathbf{q}}^r + \mathbf{\Omega}^2\,\mathbf{q}^r + \mathbf{\Gamma}^r\,\mathbf{x}^* &= \mathbf{0} \\ \ddot{\mathbf{q}}^c + \mathbf{q}^\lambda + \mathbf{\Gamma}^c\,\mathbf{x}^* &= \mathbf{0} \\ \mathbf{q}^c &= \mathbf{0} \\ \dot{\mathbf{x}}^* + \mathbf{\Sigma}\,\mathbf{x}^* + \mathbf{\Pi}^r\,\mathbf{q}^r + \mathbf{\Pi}^c\,\mathbf{q}^c + \mathbf{\Delta}^r\,\dot{\mathbf{q}}^r + \mathbf{\Delta}^c\,\dot{\mathbf{q}}^c + \mathbf{\Upsilon}^r\,\ddot{\mathbf{q}}^r + \mathbf{\Upsilon}^c\,\ddot{\mathbf{q}}^c &= \mathbf{0} \end{aligned} \tag{4.88}$$

The characteristic equation is the determinant of the following matrix:

$$\left[\begin{array}{ccc|ccc|ccc|cc} P_{\alpha_f}\,\mathbf{\Omega}^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & P_{\alpha_m}\,\mathbf{I} & \mathbf{0} & \mathbf{0} & P_{\alpha_f}\,\mathbf{\Gamma}^r & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & P_{\alpha_f}\,\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & P_{\alpha_m}\,\mathbf{I} & \mathbf{0} & P_{\alpha_f}\,\mathbf{\Gamma}^c & \mathbf{0} \\ \mathbf{0} & P_{\alpha_f}\,\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline P_1\mathbf{I} & \mathbf{0} & \mathbf{0} & -h\,\mathbf{I} & \mathbf{0} & \mathbf{0} & -P_\beta h^2\,\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & P_1\mathbf{I} & \mathbf{0} & \mathbf{0} & -h\,\mathbf{I} & \mathbf{0} & \mathbf{0} & -P_\beta h^2\,\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & P_1\mathbf{I} & \mathbf{0} & \mathbf{0} & -h\,\mathbf{I} & \mathbf{0} & \mathbf{0} & -P_\beta h^2\,\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & P_1\mathbf{I} & \mathbf{0} & \mathbf{0} & -P_\gamma h\,\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & P_1\mathbf{I} & \mathbf{0} & \mathbf{0} & -P_\gamma h\,\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & P_1\mathbf{I} & \mathbf{0} & \mathbf{0} & -P_\gamma h\,\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \hline P_{\alpha_f}\,\mathbf{\Pi}^r & P_{\alpha_f}\,\mathbf{\Pi}^c & \mathbf{0} & P_{\alpha_f}\,\mathbf{\Delta}^r & P_{\alpha_f}\,\mathbf{\Delta}^c & 0 & P_{\alpha_m}\,\mathbf{\Upsilon}^r & P_{\alpha_m}\,\mathbf{\Upsilon}^c & \mathbf{0} & P_{\alpha_f}\,\mathbf{\Sigma} & P_{\alpha_m}\,\mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & P_1\mathbf{I} & -P_\gamma h\,\mathbf{I} \end{array}\right] \tag{4.89}$$

and after lengthy but straightforward algebraic manipulations, one finds the equivalent form:

$$(P_\infty^{\mathbf{q}}(\zeta))^{2m}\,\det\left[\begin{array}{ccc|cc} P_{\alpha_f}(\zeta)\,\mathbf{\Omega}^2 & \mathbf{0} & P_{\alpha_m}(\zeta)\,\mathbf{I} & P_{\alpha_f}(\zeta)\,\mathbf{\Gamma}^r & \mathbf{0} \\ P_1(\zeta)\mathbf{I} & -h\,\mathbf{I} & -P_\beta(\zeta)h^2\,\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & P_1(\zeta)\mathbf{I} & -P_\gamma(\zeta)h\,\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \hline P_{\alpha_f}(\zeta)\,\mathbf{\Pi}^r & P_{\alpha_f}(\zeta)\,\mathbf{\Delta}^r & P_{\alpha_m}(\zeta)\,\mathbf{\Upsilon}^r & P_{\alpha_f}(\zeta)\,\mathbf{\Sigma} & P_{\alpha_m}(\zeta)\,\mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & P_1(\zeta)\mathbf{I} & -P_\gamma(\zeta)h\,\mathbf{I} \end{array}\right] = 0 \tag{4.90}$$

The determinant in (4.90) is strictly equivalent to the characteristic equation obtained in the unconstrained case (4.81). In this sense, the presence of algebraic constraints has a similar impact on the stability of the mechatronic integration scheme than in the purely mechanical case.

4.2.5 Convergence analysis

After elimination of the input and output variables, the equivalent dynamic equations can be formulated with the vector of dummy variables $\mathbf{z}$ ($\dot{\mathbf{z}} = \mathbf{x}$):

$$\mathbf{M}\,\ddot{\mathbf{q}} - \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}, t) + \mathbf{\Phi}_{\mathbf{q}}^T\,(k\,\boldsymbol{\lambda} + p\,\mathbf{\Phi}) - \mathbf{L}^{\mathbf{q}o}\,\widetilde{\mathbf{f}}^o(\dot{\mathbf{z}}, \mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, t) = \mathbf{0} \tag{4.91}$$

$$k\,\mathbf{\Phi}(\mathbf{q}) = \mathbf{0} \tag{4.92}$$

$$\ddot{\mathbf{z}} - \widetilde{\mathbf{f}}^s(\dot{\mathbf{z}}, \mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, t) = \mathbf{0} \tag{4.93}$$

For a purely mechanical system, the convergence of the generalized-α method has been assessed from the assumption of a constant mass matrix (section 3.3.4). Considering a mechatronic system, due to possible acceleration measurements, the functions $\widetilde{\mathbf{f}}^s$ and $\widetilde{\mathbf{f}}^o$ depend on $\ddot{\mathbf{q}}$. If this dependence is strongly nonlinear, we may not be able to guarantee a good convergence for the generalized-α method. As we shall see, the generalized-α method behaves in a reliable way if the dependence of $\widetilde{\mathbf{f}}^s$ and $\widetilde{\mathbf{f}}^o$ with respect to the acceleration is linear, which motivates the following technical assumption:

Assumption 4.2 *The functions* $\widetilde{\mathbf{f}}^s(\mathbf{x}, \mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, t)$ *and* $\widetilde{\mathbf{f}}^o(\mathbf{x}, \mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, t)$ *are linear in* $\ddot{\mathbf{q}}$*:*

$$\widetilde{\mathbf{f}}^s = \widetilde{\mathbf{f}}^{s^*}(\dot{\mathbf{z}}, \mathbf{q}, \dot{\mathbf{q}}, t) + \widetilde{\mathbf{f}}^s_{\ddot{\mathbf{q}}}\ \ddot{\mathbf{q}} \tag{4.94}$$

$$\widetilde{\mathbf{f}}^o = \widetilde{\mathbf{f}}^{o^*}(\dot{\mathbf{z}}, \mathbf{q}, \dot{\mathbf{q}}, t) + \widetilde{\mathbf{f}}^o_{\ddot{\mathbf{q}}}\ \ddot{\mathbf{q}} \tag{4.95}$$

where $\widetilde{\mathbf{f}}^o_{\ddot{\mathbf{q}}}$ *and* $\widetilde{\mathbf{f}}^o_{\ddot{\mathbf{q}}}$ *are constant. Moreover, the matrix* $\widetilde{\mathbf{M}} = \mathbf{M} - \mathbf{L}^{\mathbf{q}o}\ \widetilde{\mathbf{f}}^o_{\ddot{\mathbf{q}}}$ *is not singular.*

The dynamic equations can then be restated:

$$\begin{bmatrix} \widetilde{\mathbf{M}} & \mathbf{0} \\ -\widetilde{\mathbf{f}}^s_{\ddot{\mathbf{q}}} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \ddot{\mathbf{z}} \end{bmatrix} + \begin{bmatrix} \mathbf{\Phi}_{\mathbf{q}}^T\ (k\,\boldsymbol{\lambda} + p\,\mathbf{\Phi}) - \mathbf{g} - \mathbf{L}^{\mathbf{q}o}\ \widetilde{\mathbf{f}}^{o^*} \\ -\widetilde{\mathbf{f}}^{s^*} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{4.96}$$

$$\mathbf{\Phi}(\mathbf{q}) = \mathbf{0} \tag{4.97}$$

These equations have the same structure than the equations of a purely mechanical system. Hence, the same convergence properties are observed, and we immediately conclude that the local truncation error is still of order 2.

Remark 4.3 *Assumption 4.2 is satisfied for any linear control system and any control system without acceleration measurement, and those categories cover many practical situations.*

Remark 4.4 *If the functions* $\widetilde{\mathbf{f}}^s$ *and* $\widetilde{\mathbf{f}}^o$ *are not linear but only affine with respect to* $\ddot{\mathbf{q}}$ *(*i.e.*, in equations* (4.94) *and* (4.95)*,* $\widetilde{\mathbf{f}}^s_{\ddot{\mathbf{q}}}$ *and* $\widetilde{\mathbf{f}}^o_{\ddot{\mathbf{q}}}$ *are not constant), the arguments described in section 3.3.4 about the consequences of a non-constant mass matrix convince us that the local truncation error might increase by one order. For instance, such a situation arises when acceleration signals* $\mathbf{w}^{ma}$ *are obtained from a 3-axis accelerometer fixed on a moving body. The measurements are the*

absolute accelerations in body axes, and they are related to the accelerations in inertial axes $\ddot{\mathbf{q}}$ *by the rotation matrix of the body* $\mathbf{R}$*:*

$$\mathbf{w}^{ma} = \mathbf{R}^T \, \ddot{\mathbf{q}} \tag{4.98}$$

In case of very fast motion, the variations of $\mathbf{R}$ *may not be negligible over each time-step, leading to a small loss in accuracy.*

Remark 4.5 *Control systems involving non-affine acceleration feedbacks are quite unusual for control practitioners, so that we should not be afraid about the restrictions involved by Assumption 4.2.*

4.2.6 Optimal choice for the algorithmic parameters

The previous results prove that the time-integration algorithm for mechatronic systems inherits many properties from the purely "mechanical" and "state-space" cases. The condition

$$\gamma = \frac{1}{2} + \alpha_{fm} \tag{4.99}$$

leads to second-order accuracy for the coupled system. For given values of α_f and α_m, the spurious roots of both subsystems are equal:

$$\zeta_1^{\mathbf{q}} = \zeta_1^{\mathbf{x}} = \zeta_1 = \frac{-\alpha_f}{1-\alpha_f} \tag{4.100}$$

whereas the principal roots are different (Figure 4.7):

$$\zeta_{2,3}^{\mathbf{q}} = -\frac{1-\alpha_{fm}}{1+\alpha_{fm}} \quad \neq \quad \zeta_2^{\mathbf{x}} = -\frac{1-2\alpha_{fm}}{1+2\alpha_{fm}} \tag{4.101}$$

This explains why the optimal choices of α_f and α_m are different for a mechanical model and a state-space model (compare formulae (3.30) and (4.49)).

According to Chung and Hulbert [CH93], the optimal parameters for the simulation of the *mechanical equations* are obtained when $|\zeta_1| = |\zeta_{2,3}^{\mathbf{q}}|$. From Figure 4.7, we then conclude

$$|\zeta_2^{\mathbf{x}}| < |\zeta_1| = \rho_\infty^{\mathbf{q}} = \rho_\infty^{\mathbf{x}} \tag{4.102}$$

for $\alpha_{fm} < \frac{\sqrt{2}}{2}$. Since the modulus of the principle root $|\zeta_2^{\mathbf{x}}|$ is lower than the modulus of the spurious root, this design is non-optimal for the state variables.

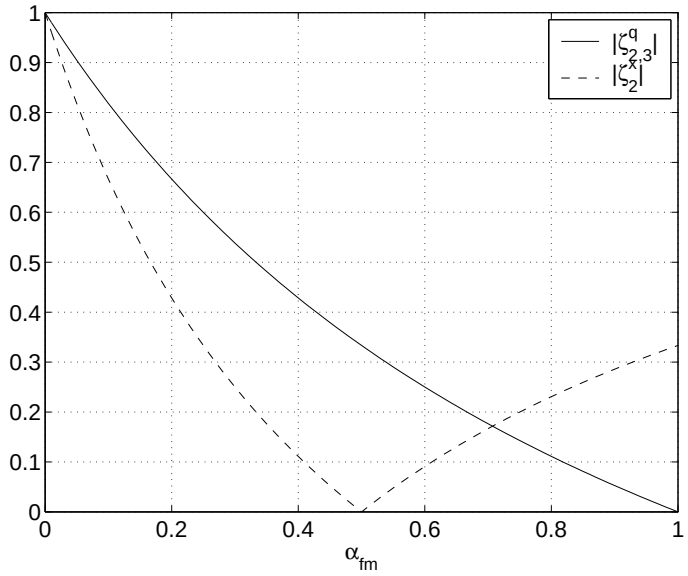


Figure 4.7: Evolution of the roots with respect to the parameter α_{fm}.

Conversely, according to Jansen *et al.* [JWH00], the optimal parameters for the *state equations* are obtained when $|\zeta_1| = |\zeta_2^{\mathbf{x}}|$. From Figure 4.7, we conclude

$$|\zeta_{2,3}^{\mathbf{q}}| = \rho_\infty^{\mathbf{q}} > |\zeta_1| = \rho_\infty^{\mathbf{x}} \tag{4.103}$$

for $\alpha_{fm} < \frac{\sqrt{2}}{2}$. This design is not optimal for the mechanical variables. In particular, the spectral radius associated with the mechanical variables is higher than the spectral radius selected for the state variables.

The Hilber-Hughes-Taylor algorithm [HHT77] is obtained for $\alpha_m = 0$, and a positive value for α_f. The spurious root should satisfy the condition:

$$|\zeta_2^{\mathbf{x}}| > |\zeta_1| \qquad \text{and} \qquad |\zeta_{2,3}^{\mathbf{q}}| > |\zeta_1| \tag{4.104}$$

which imposes the design constraint $\alpha_f < \frac{1}{4}$. In this case, the spectral radius associated with the mechanical variables, is always higher than the spectral radius associated with the state variables: $\rho_\infty^{\mathbf{q}} > \rho_\infty^{\mathbf{x}}$.

Hence, the optimal choice of the algorithmic parameters for the coupled set of equations is not trivial; in the numerical applications presented in this chapter, the Hilber-Hughes-Taylor method is systematically applied.

4.3 Systems with discontinuous dynamics

A hybrid system involves both continuous dynamic variables and discrete events. After a discrete event, it is well-known that a restart procedure is necessary to carry on the simulation. In this section, a restart procedure is developed for the mechatronic simulation algorithm.

In this work, we have in mind two types of discontinuous behavior:

- the discrete time behavior of a micro-controller or a micro-processor, where explicit jumps are observed at the output $\mathbf{f}^o(t^+_{n+1}) \neq \mathbf{f}^o(t^-_{n+1})$,

- the saturation effect of a software control law or a hardware component, which is mathematically formulated as a change in the dynamic behavior $\mathbf{f}^s(\mathbf{u}, \mathbf{x}, t^+_{n+1}) \neq \mathbf{f}^s(\mathbf{u}, \mathbf{x}, t^-_{n+1})$ or $\mathbf{f}^o(\mathbf{u}, \mathbf{x}, t^+_{n+1}) \neq \mathbf{f}^o(\mathbf{u}, \mathbf{x}, t^-_{n+1})$.

Those discontinuities may produce discontinuities in the generalized forces. According to Newton's second law of motion, the discontinuities are thereby transmitted to the accelerations $\ddot{\mathbf{q}}$, and to the reaction forces represented by the Lagrange multipliers $\boldsymbol{\lambda}$ [BDG02, BDG03]. The state rates $\dot{\mathbf{x}}$ and the algebraic variables $\mathbf{y}$ are also directly affected:

$$\ddot{\mathbf{q}}^-_{n+1} \neq \ddot{\mathbf{q}}^+_{n+1}, \qquad \boldsymbol{\lambda}^-_{n+1} \neq \boldsymbol{\lambda}^+_{n+1}, \qquad \dot{\mathbf{x}}^-_{n+1} \neq \dot{\mathbf{x}}^+_{n+1}, \qquad \mathbf{y}^-_{n+1} \neq \mathbf{y}^+_{n+1} \tag{4.105}$$

Temporarily, we exclude impulse phenomena, so that the energy is continuous, as well as $\mathbf{q}$, $\dot{\mathbf{q}}$ and $\mathbf{x}$:

$$\mathbf{q}^-_{n+1} = \mathbf{q}^+_{n+1}, \qquad \dot{\mathbf{q}}^-_{n+1} = \dot{\mathbf{q}}^+_{n+1}, \qquad \mathbf{x}^-_{n+1} = \mathbf{x}^+_{n+1} \tag{4.106}$$

The integration scheme developed in the previous section was able to predict efficiently the values at time t^-_{n+1}. Since the integration formulae rely on a Taylor expansion of the dynamic variables, they are not applicable over the discontinuous transition from t^-_{n+1} to t^+_{n+1}, and the correction for $\ddot{\mathbf{q}}$, $\boldsymbol{\lambda}$, $\dot{\mathbf{x}}$ and $\mathbf{y}$ should be performed *independently* of $\mathbf{q}$, $\dot{\mathbf{q}}$ and $\mathbf{x}$, which are kept constant.

This correction can be seen as an integration restart procedure, and we know from earlier convergence results (see section 3.3.4) that the initial conditions at

time t_{n+1}^+ should satisfy the residual equations:

$$\mathbf{M}\,\ddot{\mathbf{q}}_{n+1}^+ + \boldsymbol{\Phi}_{\mathbf{q}}^T\,(k\,\boldsymbol{\lambda}_{n+1}^+ + p\,\boldsymbol{\Phi}) - \mathbf{g}(\mathbf{q}_{n+1}^+, \dot{\mathbf{q}}_{n+1}^+, t) - \mathbf{L}^{\mathbf{q}o}\,\mathbf{y}_{n+1}^+ = \mathbf{0} \tag{4.107}$$

$$k\,\boldsymbol{\Phi}(\mathbf{q}_{n+1}^+) = \mathbf{0} \tag{4.108}$$

$$\dot{\mathbf{x}}_{n+1}^+ - \mathbf{f}^s(\mathbf{u}_{n+1}^+, \mathbf{x}_{n+1}^+, t_{n+1}^+) = \mathbf{0} \tag{4.109}$$

$$\mathbf{y}_{n+1}^+ - \mathbf{f}^o(\mathbf{u}_{n+1}^+, \mathbf{x}_{n+1}^+, t_{n+1}^+) = \mathbf{0} \tag{4.110}$$

$$\mathbf{u}_{n+1}^+ - \mathbf{L}^{i\mathbf{q}}\,\mathbf{q}_{n+1}^+ - \mathbf{L}^{i\dot{\mathbf{q}}}\,\dot{\mathbf{q}}_{n+1}^+ - \mathbf{L}^{i\ddot{\mathbf{q}}}\,\ddot{\mathbf{q}}_{n+1}^+ - \mathbf{L}^{io}\,\mathbf{y}_{n+1}^+ = \mathbf{0} \tag{4.111}$$

otherwise, the order of convergence would be deeply affected. $\mathbf{q}_{n+1}^+$, $\dot{\mathbf{q}}_{n+1}^+$ and $\mathbf{x}_{n+1}^+$ are imposed by (4.106), and these equations can be seen as nonlinear algebraic equations for $\ddot{\mathbf{q}}_{n+1}^+$, $\boldsymbol{\lambda}_{n+1}^+$, $\dot{\mathbf{x}}_{n+1}^+$, $\mathbf{y}_{n+1}^+$. They can be solved using a Newton-Raphson procedure with the initial values given at time t_{n+1}^-. However, since the constraint equation does not involve any of these unknowns, the system is ill-defined. In order to overcome this problem, the constraints should be formulated at the acceleration level:

$$k\,\ddot{\boldsymbol{\Phi}} = k\,\dot{\boldsymbol{\Phi}}_{\mathbf{q}}\,\dot{\mathbf{q}} + k\,\boldsymbol{\Phi}_{\mathbf{q}}\,\ddot{\mathbf{q}} = \mathbf{0} \tag{4.112}$$

with $\dot{\boldsymbol{\Phi}}_{\mathbf{q}} = \dot{\boldsymbol{\Phi}}_{\mathbf{q}}(\mathbf{q}, \dot{\mathbf{q}})$ and $\boldsymbol{\Phi}_{\mathbf{q}} = \boldsymbol{\Phi}_{\mathbf{q}}(\mathbf{q})$. If this hidden constraint is satisfied at time t_{n+1}^-, it implies:

$$k\,\boldsymbol{\Phi}_{\mathbf{q}}\,(\ddot{\mathbf{q}}_{n+1}^+ - \ddot{\mathbf{q}}_{n+1}^-) = \mathbf{0} \tag{4.113}$$

which may replace equation (4.108) in order to obtain a nonsingular system[1]. The linearized equations follow:

$$\begin{bmatrix} \mathbf{M} & k\,\boldsymbol{\Phi}_{\mathbf{q}}^T & \mathbf{0} & -\mathbf{L}^{\mathbf{q}o} \\ k\,\boldsymbol{\Phi}_{\mathbf{q}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & -\mathbf{f}_{\mathbf{u}}^s\,\mathbf{L}^{io} \\ -\mathbf{L}^{o\ddot{\mathbf{q}}} & \mathbf{0} & \mathbf{0} & \mathbf{I} - \mathbf{f}_{\mathbf{u}}^o\,\mathbf{L}^{io} \end{bmatrix} \begin{bmatrix} \triangle\ddot{\mathbf{q}} \\ \triangle\boldsymbol{\lambda} \\ \triangle\dot{\mathbf{x}} \\ \triangle\mathbf{y} \end{bmatrix} = \begin{bmatrix} -\mathbf{res}_k^{\mathbf{q}} \\ -\mathbf{res}_k^{\boldsymbol{\Phi}} \\ -\mathbf{res}_k^s \\ -\mathbf{res}_k^o \end{bmatrix} \tag{4.114}$$

A summary of the integration algorithm is given in Figure 4.8. It is assumed that the discrete transitions occur at the nodes of the time grid. If the discontinuities are caused by a sampled system, and if a fixed time-step strategy is adopted, this means that the sampling period T^s should be a multiple of the time-step h:

$$h = \frac{T^s}{p} \tag{4.115}$$

[1]The mechanical equations are linear in $\ddot{\mathbf{q}}$, $\boldsymbol{\lambda}$ and $\mathbf{y}$, thus, if the control system is also linear, one Newton iteration yields convergence.

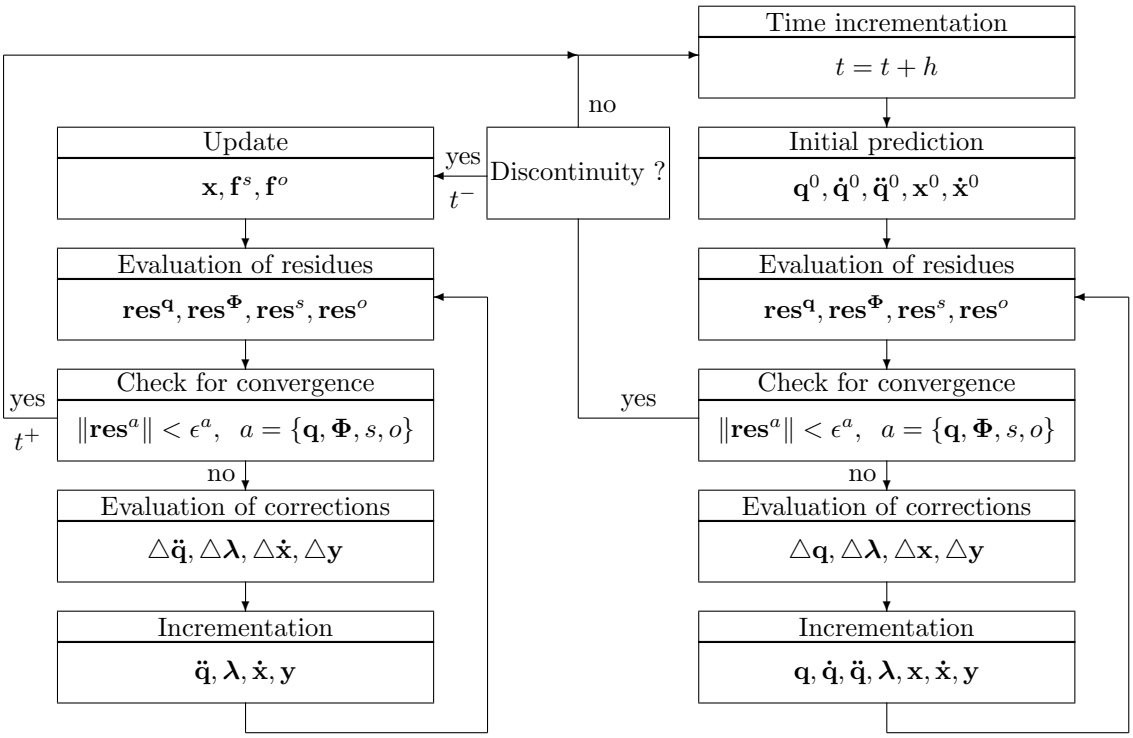


Figure 4.8: Integration algorithm for hybrid systems.

where p is an integer. In a more general context, a variable step-size time-integration strategy should be combined with an event detection algorithm in order to catch exactly the instant of the transition.

4.3.1 Impulse phenomena

In the previous development, the case of impulsive mechanical loading has been excluded. However, a slight adaptation of the algorithm is sufficient to cover this case, as described in the following. Since discontinuities now appear at the velocity level:

$$\dot{\mathbf{q}}_{n+1}^- \neq \dot{\mathbf{q}}_{n+1}^+ \tag{4.116}$$

the values of $\dot{\mathbf{q}}$ should simply be updated when the impulse is detected, afterwards the correction algorithm presented in Figure 4.8 is applicable. Therefore, the critical point is to compute $\dot{\mathbf{q}}_{n+1}^+$.

If $\mathbf{g}_e$ denotes the generalized impact forces, their integral effect $\mathbf{p}_e$ is defined

by

$$\mathbf{p}_e = \int_{t^-}^{t^+} \mathbf{g}_e \, dt \tag{4.117}$$

The generalized velocities satisfy the classical impact equations, as described by De Jalón and Bayo [DJB94]:

$$\begin{bmatrix} \mathbf{M} & \mathbf{\Phi}_\mathbf{q}^T \\ \mathbf{\Phi}_\mathbf{q} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}_{n+1}^+ - \dot{\mathbf{q}}_{n+1}^- \\ \boldsymbol{\lambda}_p \end{bmatrix} = \begin{bmatrix} \mathbf{p}_e \\ \mathbf{0} \end{bmatrix} \tag{4.118}$$

where $\boldsymbol{\lambda}_p$ are Lagrange multipliers that are related to the internal impact forces (impact reactions). The first equation represents the conservation of momentum, and the second equation states that the velocity increment has to fulfill the homogeneous velocity constraint equations.

The computation of $\dot{\mathbf{q}}_{n+1}^+$ requires the knowledge of $\mathbf{p}_e$, which is not always a trivial problem. If the impulse forces result from an impact between bodies, $\mathbf{p}_e$ can be estimated using a physical model of their contact (*e.g.* an elastic or a plastic model).

4.4 Implementation issues

The simulation concepts hereby developed for mechatronic systems have been implemented in the Oofelie C++ Finite Element software [CKG94]. The reason is that the object oriented technology is very suitable to build a modular, reusable and efficient code. Let us briefly discuss the implementation at the level of the kernel and of the element library.

At the kernel level, the dof concept is naturally extended to represent input, state and output variables, and the element concept to represent a generic block of a block diagram model. All block diagram elements inherit from a few abstract classes, such as `System`, `ContinuousSystem` and `DiscontinuousSystem`, themselves inheriting from the standard `Element` class (see Figure 4.9). Those abstract classes define

- the generic interface of the blocks,
- the concepts of input, output and state variables,
- the concepts of output and state equations, and their connection with the Finite Element assembly procedure.

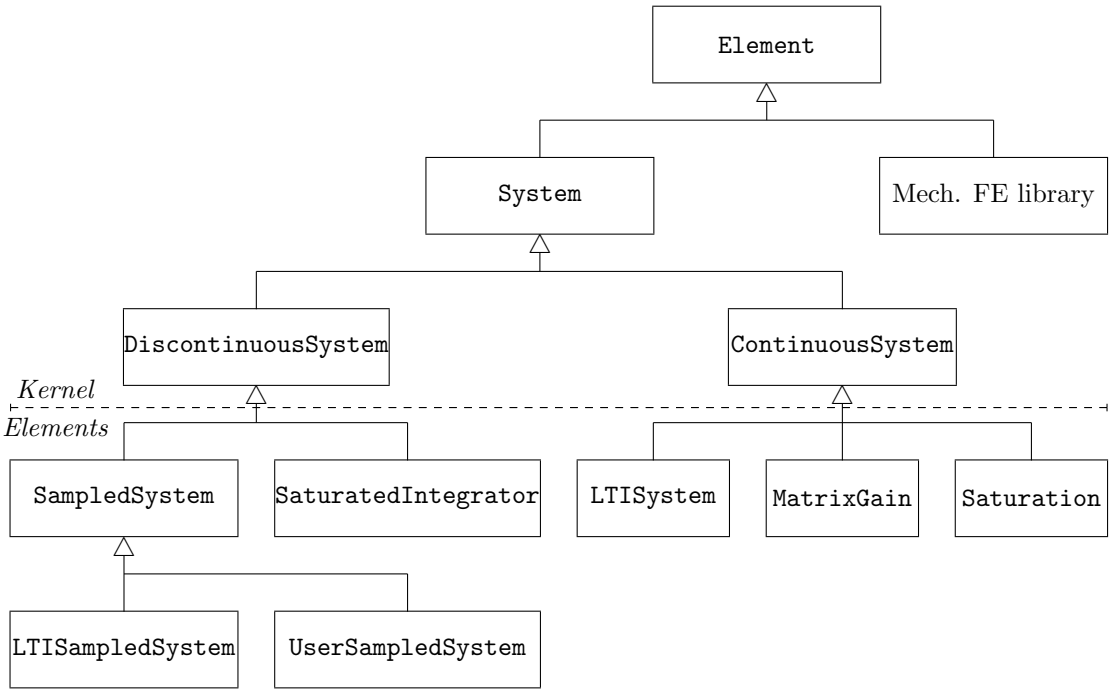


Figure 4.9: Element library: class diagram. LTI is an acronym for Linear Time Invariant.

A simplified view of the library of control elements appears in Figure 4.9. The developments realized in the kernel make the programming effort for an element minimal, since only the functions $\mathbf{f}^s$ and $\mathbf{f}^o$, and their sensitivity matrices have to be defined (the implementation of the sensitivities can be avoided using a finite difference algorithm).

One may not be surprised to find the `SaturatedIntegrator` among the discontinuous elements, and the `Saturation` among continuous elements:

- the `SaturatedIntegrator` is characterized by one state variable x which integrates the output until the saturation value x^{max}:

$$\dot{x} = u, \quad \text{while} \quad x < x^{max} \tag{4.119}$$

At the saturation the state equation becomes:

$$\dot{x} = 0, \quad x = x^{max}, \quad \text{while} \quad u > 0 \tag{4.120}$$

The discontinuity of $\dot{x}$ requires a restart procedure,

- the `Saturation` element is characterized by one output variable, which is equal to the input variable until the saturation value y^{max}:

$$y = u, \quad \text{while} \quad u < y^{max} \tag{4.121}$$

After saturation, the output equation becomes:

$$y = y^{max}, \quad \text{while} \quad u > y^{max} \tag{4.122}$$

and y is continuous.

4.5 Applications

Three examples with increasing complexity are considered: a rigid four-bar mechanism, a Scara robot, and a vehicle with semi-active suspensions.

4.5.1 Rigid four-bar mechanism

Let us consider the rigid four-bar mechanism illustrated in Figure 4.10. It is subject to external loads $\mathbf{g}^{ext}$, the lower left hinge is actuated, and the value of the torque τ is defined by a proportional integral derivative (PID) control law[2] (see Figure 4.11). An analog and a digital implementation of the PID law will be analyzed.

The analog PID controller is represented by the scalar state equations:

$$\dot{x} = \theta - \theta^{ref} \tag{4.123}$$

$$y = -P\,(\theta - \theta^{ref}) - D\,\dot{\theta} - I\,x \tag{4.124}$$

The digital PID controller generates a piecewise constant output

$$y = y_k \qquad \forall\, t \in [t_k, t_{k+1}] \tag{4.125}$$

with the update equation

$$x_{k+1} = x_k + T^s\,(\theta_{k+1} - \theta^{ref}_{k+1}) \tag{4.126}$$

$$y_{k+1} = -P\,(\theta_{k+1} - \theta^{ref}_{k+1}) - D\,\dot{\theta}_{k+1} - I\,x_{k+1} \tag{4.127}$$

[2]We consider here a regulation problem, so that the derivative action is simply established from the measurement rate $\dot{\theta}$, and not from the error rate $\dot{e}$.

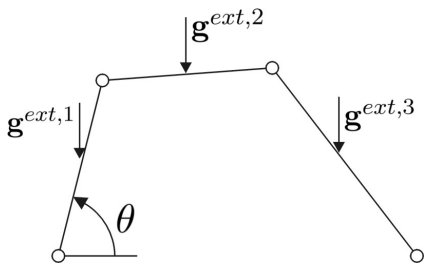


Figure 4.10: Four-bar mechanism.

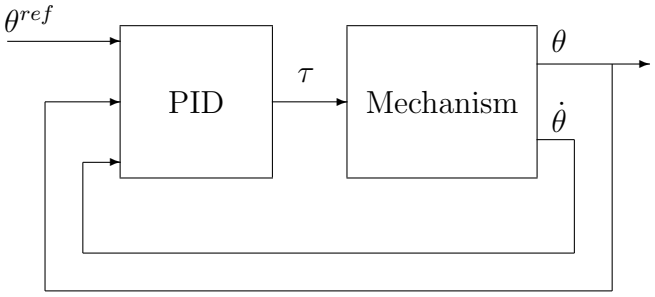


Figure 4.11: PID control of a four-bar mechanism.

with the sampling period $T^s = 0.01\ s$. The discrete state x_k is not treated as a dof in the simulation, it is simply updated at every sampling period. The maximal value of the time-step is equal to the sampling period: $h^{(max)} = T^s$.

In this example, the mechanism is initially at rest ($\mathbf{q}_0 = \mathbf{q}_{init}$, $\dot{\mathbf{q}}_0 = 0$), and the time variation of the external loads and reference inputs is plotted in Figure 4.12. We may notice that the initial values $\ddot{\mathbf{q}} = \mathbf{0}$ and $\dot{x} = 0$ (for the analog control) are consistent with those initial conditions.

The concise and high level textual model definition in Oofelie is illustrated in Figure 4.13. For the digital case, the syntax is similar, the user should simply further specify the sampling period.

Simulation results

A small numerical damping is necessary to avoid high frequency oscillations due to the mechanical algebraic constraints, and we have selected $\alpha_f = 0.05$, $\alpha_m = 0$ (Hilber-Hughes-Taylor algorithm).

A first simulation is realized with a time-step $h^{(1)} = h^{(max)} = 0.01\ s$, and the

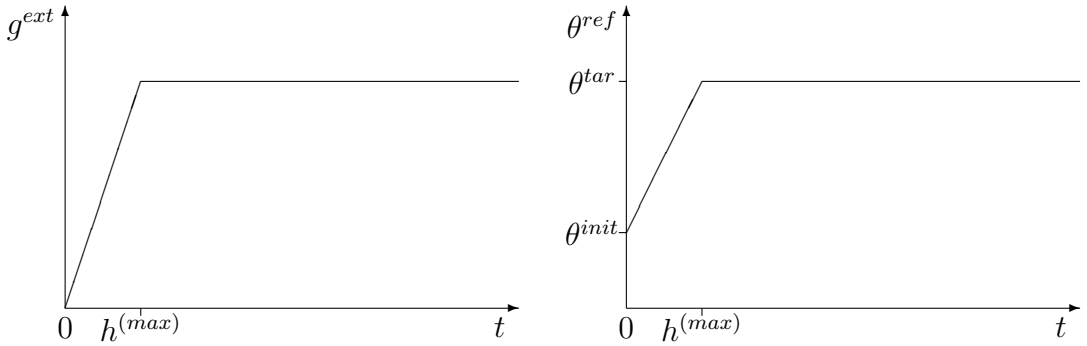


Figure 4.12: Four-bar mechanism with PID control: external loads and reference input. $h^{(max)} = 0.01\,s$.

```
ElemSet.define(PID,ContinuousLTISystem_E,3,1,1);
ElemSet[PID].addInput(ThetaRefNode,GenDisp);
ElemSet[PID].addInput(HingeAngleNode,GenDisp);
ElemSet[PID].addInput(HingeAngleNode,GenVel);
ElemSet[PID].addOutput(TorqueNode);
ElemSet[PID].setStateSpaceMatrices(A,B,C,D);
```

Figure 4.13: Oofelie input data file - description of an analog PID controller (3 inputs, 1 state and 1 output). `PID` is the element number; `ThetaRefNode, HingeAngleNode, TorqueNode` are node numbers; `GenDisp, GenVel` mean "generalized displacement" and "generalized velocity" respectively; and `A,B,C,D` are the state-space matrices associated with equations (4.123) and (4.124).

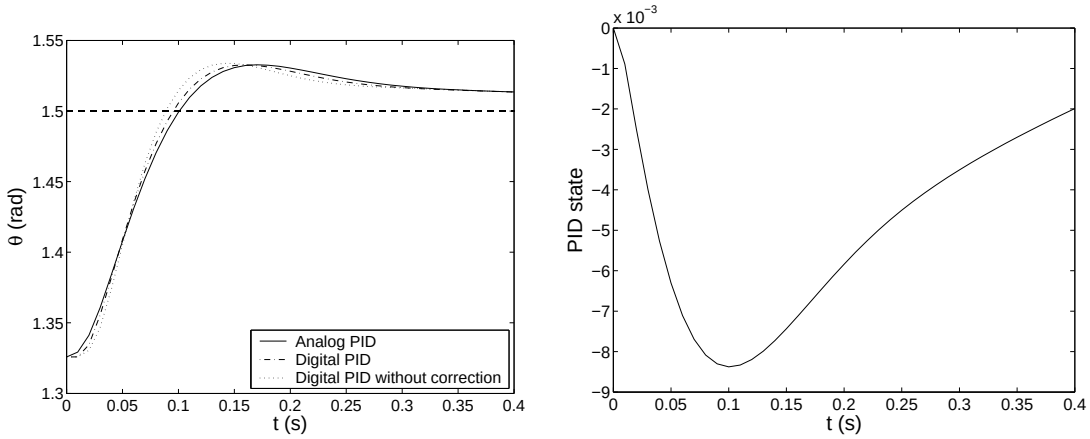


Figure 4.14: Simulation of a four-bar mechanism with PID control.

results are presented in Figure 4.14. A noticeable difference is observed whether the system is controlled by an analog or a digital controller. For the digital control case, a simplified integration scheme has also been tested, where the specific correction step after discontinuities is omitted. We observe that the error in the simplified algorithm and the sampling effect have the same order of magnitude, and we conclude that the correction step is essential for the accurate representation of the sampling effect.

In order to analyze the influence of a time-step reduction, five different values have been considered:

$$h^{(p)} = \frac{h^{(max)}}{2^{p-1}} \qquad p = \{1, ..., 5\} \tag{4.128}$$

In Figure 4.15, the results obtained with $h^{(1)}$ and $h^{(5)}$ are very close to each other for both the analog and the digital controller. However, for the digital controller, the simplified algorithm without discontinuity correction leads to important errors, especially for $h^{(1)}$.

Assuming that the results obtained with $h^{(5)}$ are close to the exact solution, we define the error for a dynamic variable as the deviation:

$$\sigma(h^{(p)}) = \sqrt{\frac{1}{n} \sum_{n} (q_n^{(p)} - q_n^{(5)})^2} \qquad p = \{1, ..., 4\} \tag{4.129}$$

where $q_n^{(p)}$ is the value of the dynamic variable predicted with a time-step $h^{(p)}$ at time t_n. The evolution of σ with respect to the time-step is plotted in Figure 4.16.

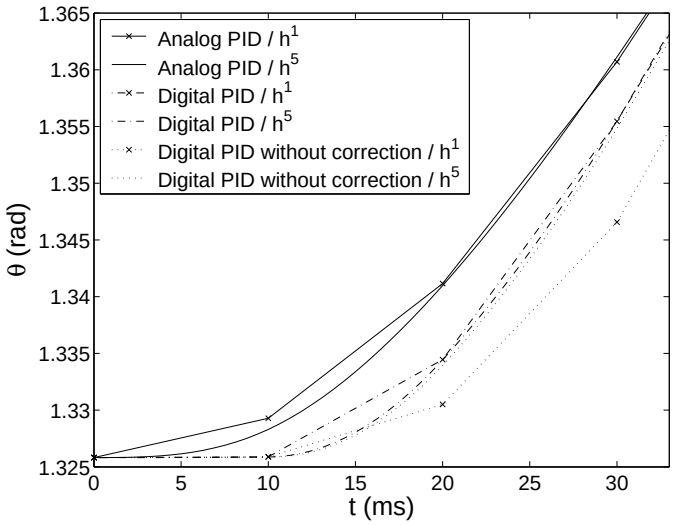


Figure 4.15: Four-bar mechanism with PID control - influence of the time-step ($h^{(1)} = 10\,ms,\ h^{(5)} = 0.625\,ms$).

The expected second-order convergence is observed in all cases, excepted for the simplified algorithm which exhibits a poor first-order convergence. Therefore, the simplified algorithm is definitely not acceptable for the simulation of a mechatronic system with a digital controller. *For the corrected algorithm, we conclude that the order of convergence predicted by the theory is verified.*

4.5.2 Scara robot

In order to further validate our approach, let us consider a benchmark proposed in the Eurosim framework in order to compare the capabilities of various software [For98, Sch98, Eck99, Sas04]. The system under consideration is a Scara robot (see Figure 4.17), and the model accounts for the dynamics of the joint DC motors, and for their PD control law. The first two joints are revolute with vertical axes, and a third prismatic joint yields motion in the vertical direction. The electrical current in each motor is controlled independently according to a collocated PD law. Figure 4.18 presents a block diagram model for the control law and the actuator associated with one joint. The PD control law computes the reference voltage U^{ref}, and the applied voltage U^a is limited to maximal values:

$$U^a = U^{ref} \qquad \forall\, |U^a| \leq U^{max} \tag{4.130}$$

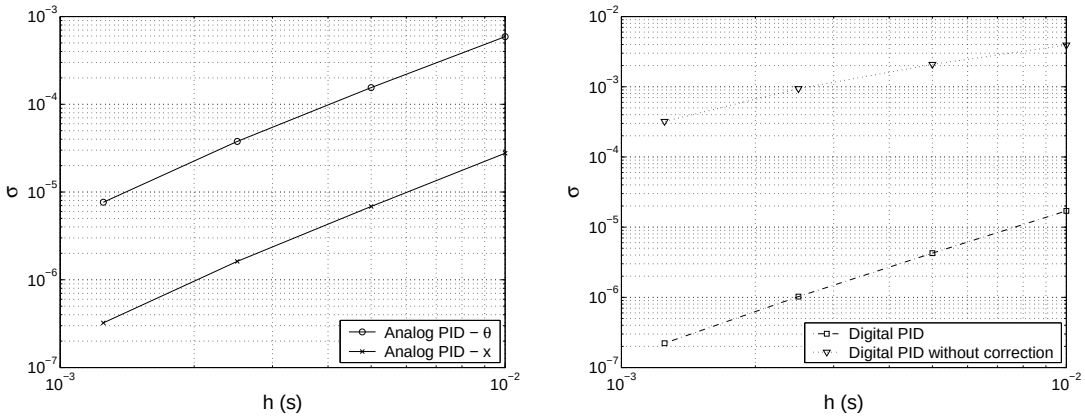


Figure 4.16: Four-bar mechanism with PID control - convergence analysis. On the left, the error for the hinge angle and the controller state (analog PID). On the right, the error for the hinge angle (digital PID).

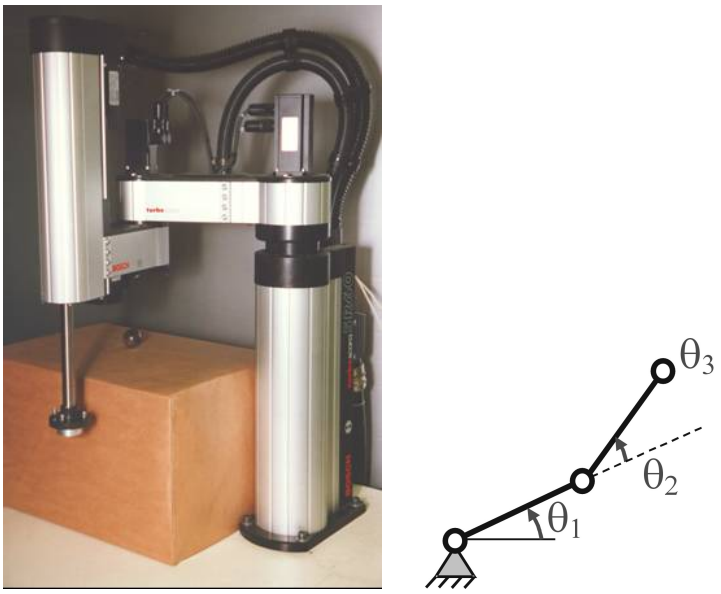


Figure 4.17: Scara robot.

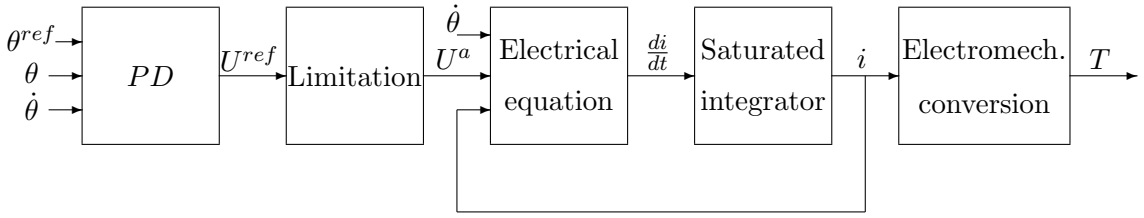


Figure 4.18: Scara robot - Controller and motor associated with one joint.

The electrical current satisfies the following equation

$$L \frac{di}{dt} = U^a - U^{em} - R\, i \qquad \forall\, |i| \leq i^{max} \tag{4.131}$$

The electrical current is also limited to a maximal value, and U^{em} is the electromotive voltage proportional to the joint velocity:

$$U^{em} = k^{em}\, \dot{\theta} \tag{4.132}$$

The electromechanical equation defines the relation between the torque and the current intensity:

$$T = k^t\, i \tag{4.133}$$

k^{em} and k^t are two parameters of the electrical motor.

The behavior of the system is simulated when a step reference command is applied at $t = 0\ s$ (Figure 4.19):

Initial position	Target position
$\theta_1 = 0\ rad$	$\theta_1 = 2\ rad$
$\theta_2 = 0\ rad$	$\theta_2 = 2\ rad$
$\theta_3 = 0\ m$	$\theta_3 = 0.3\ m$

(4.134)

The results of the simulation obtained for a time-step $h = 0.001$ s are presented in Figures 4.19 and 4.20, and are fully equivalent to the results obtained by other researchers [For98, Sch98, Eck99, Sas04]. Even though the mechanical motion is rather smooth, the electrical current is subject to significant transient phenomena at the beginning and at the end of the motion. In particular, the simulation predicts saturation effects.

A digital implementation of the control law has also been simulated with a sampling period $T^s = 5\ ms$, leading to the results in Figure 4.21. In motor 1,

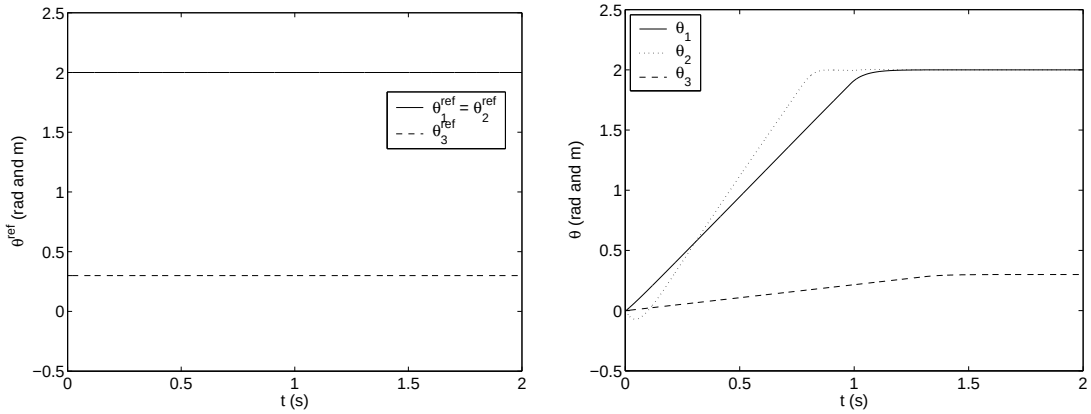


Figure 4.19: Scara robot, joint references (step signal) and joint coordinates.

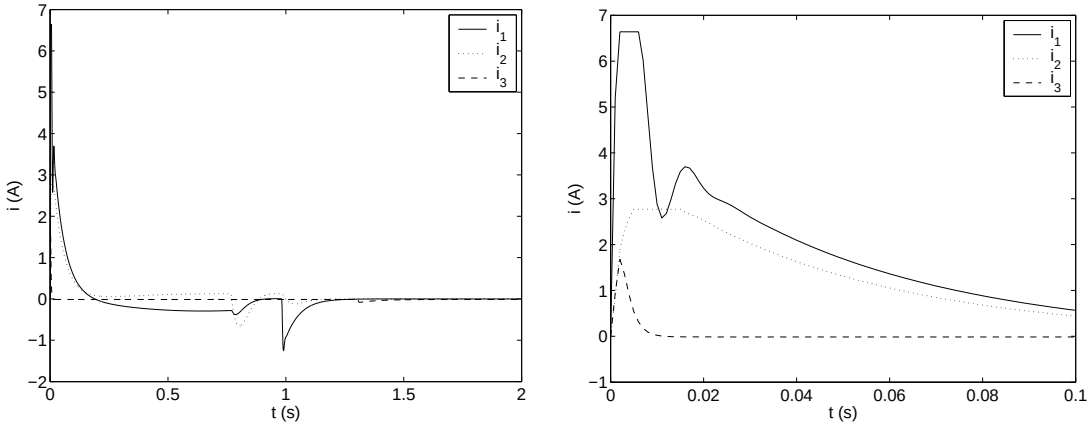


Figure 4.20: Scara robot - electrical currents. The picture on the right is a zoom.

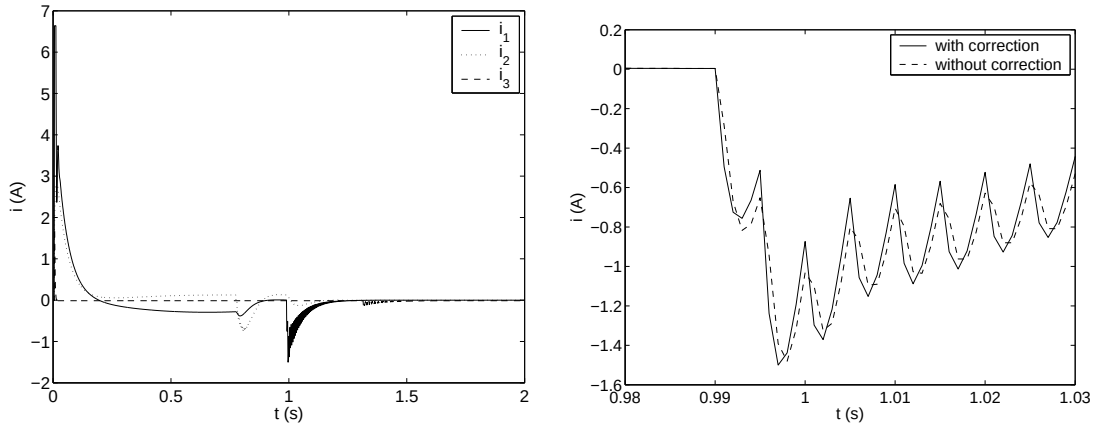


Figure 4.21: Scara robot with sampled controller - electrical current in motor 1. The picture on the right is a zoom centered on $t = 1s$.

a high-frequency electromechanical pole involving the electromotive voltage is excited at the end of the motion. According to a linearized analysis, if J^{eq} denotes the equivalent inertia associated with the motion of joint 1, the pulsation of this pole is:

$$\omega = \sqrt{\frac{k^{em}\, k^t}{J^{eq}\, L}} \tag{4.135}$$

The resulting order of magnitude is in agreement with the simulation results. Actually, due to the resistor, this pole is also affected by a damping effect. Again, the simplified simulation method, without discontinuity correction does not lead to accurate results.

From this example, we conclude that our simulation method is able to deal systematically and reliably with the coupled electromechanical equations of a mechatronic system. The block diagram model is modular and convenient for the dynamic description of the controller and the actuators. Most methods presented by other authors were based on an adaptive time-stepping method, which is a strong advantage over our current fixed time-step implementation. It is therefore hopeless to attempt any comparison with respect to the computational efficiency, and this weakness of our software should be addressed in future investigations. However, the decisive advantage of our approach certainly comes from its generality, in particular its ability to deal efficiently and accurately with complex parallel topologies and mechanical deformations; those potentialities are not really highlighted in this simple example. The objective of the next section is to

demonstrate the relevance of the integrated simulation method for a mechanism with a far more complex topology, subject to the action of a strongly nonlinear control system.

4.5.3 Vehicle semi-active suspension

The design of a modern car involves an increasing number of active components, *e.g.* active suspension, anti-lock brake system (ABS), or electronic stability program (ESP). Therefore, in order to predict the car performances, the dynamics of those active components has to be considered in the design.

The dynamic simulation of a car equipped with a semi-active suspension is considered here. This benchmark has been identified in the framework of the Belgian Inter-University Attraction Pole on Advanced Mechatronic Systems (AMS-IAP5/06)[3]. The objective of the semi-active suspension demonstrator is to develop new concepts related with the modeling, control and optimization of mechatronic systems. Our personal contribution to this project concerns the *integrated simulation* of the full car including the actuators and controller dynamics, using the methodology developed in this work. The various components that have been considered in the multidisciplinary model of the car are described in the following sections. We believe that the resulting simulation model is a relevant basis for the numerical optimization of the suspension, which will be addressed in future research.

Principle of a semi-active suspension

A passive suspension has fixed stiffness and damping characteristics determined by their design. Depending on the road conditions, an adaptation of those characteristics is desirable to optimize the comfort and the road handling, which is possible if the passive shock absorber is replaced by an active or semi-active actuator. Active suspensions lead to highly controllable systems, but they require an external power source (*e.g.* a hydraulic pump), which makes them costly, complex and less reliable. Using a semi-active actuator in parallel with a passive spring, a

[3]Website of the AMS-IAP5/06: http://www.mech.kuleuven.ac.be/pma/project/ams
This project is sponsored by the Belgian state, Prime Minister's office, Science Policy Programming.

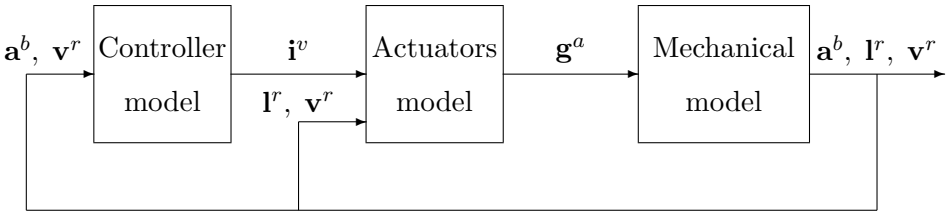


Figure 4.22: Mechatronic model of the car equipped with a semi-active suspension. $\mathbf{a}^b$ is the vector of car-body accelerations, $\mathbf{l}^r$ is the vector of rattle extensions, $\mathbf{v}^r = \dot{\mathbf{l}}^r$ is the vector of rattle velocities, $\mathbf{i}^v$ is the vector of electrical currents, and $\mathbf{g}^a$ is the vector of damper forces.

Figure 4.23: Audi A6. On the right, instrumentation with corner accelerometers.

control action is obtained by varying the restriction in current controlled valves, or by changing the viscosity of a magneto-rheological fluid. Therefore, no external power is introduced in the system, which makes the design less complicated and safer.

In our case, semi-active hydraulic actuators with current controlled valves are considered. The model of the mechatronic system is composed of a mechanical model, a model of the actuators, and a model of the controller, as illustrated in Figure 4.22.

Mechanical model

The car is an Audi A6 (see Figure 4.23), and its rigid-body model includes the following components, illustrated in Figure 4.24:

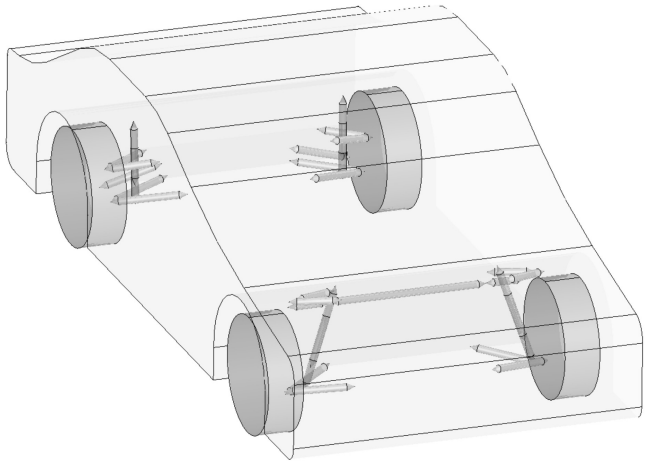


Figure 4.24: Mechanical model of the car.

- the car-body,
- the suspension mechanisms (including the passive springs),
- the slider-crank direction mechanism for the front wheels,
- the wheel model for the lateral force, the vertical force and the yaw torque.

The current model involves about 600 mechanical dofs, and it could be extended to include the stiffness of the suspension bushings, the flexibility of the chassis, and a longitudinal model for the wheels. However, the current model is sufficient to demonstrate the potentialities of the integrated simulation tool.

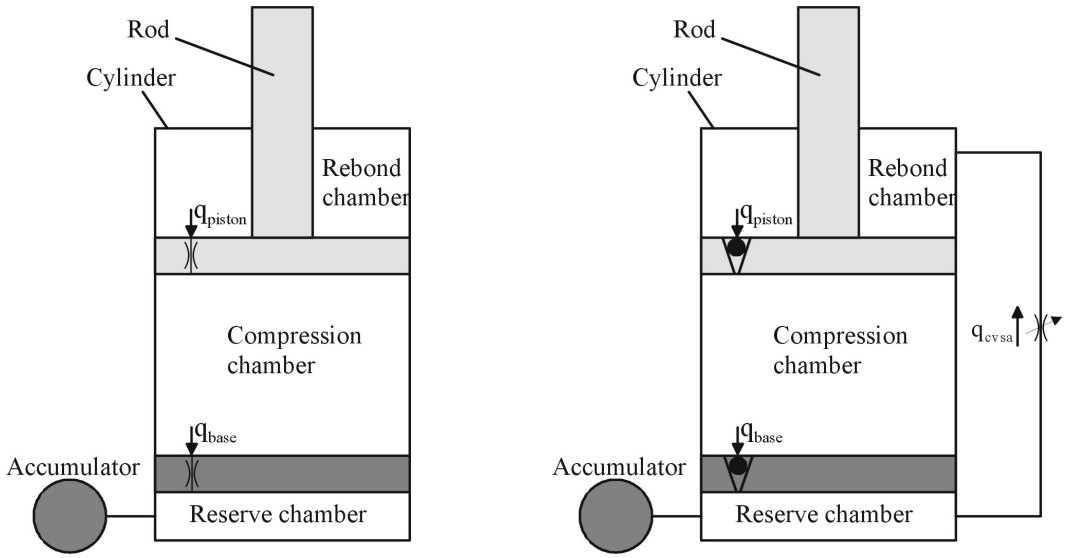
Actuator model

A systematic description of the actuator model and of the controller is presented by Lauwerys *et al.* [LSS04].

Figure 4.25 compares a passive and a semi-active shock absorber. The passive absorber contains two passive valves, restricting the oil flow from one chamber to the other. Due to the motion of the rod in and out of the cylinder, the variation of the total volume v^{tot} available for the oil is:

$$\dot{v}^{tot} = \dot{v}^{reb} + \dot{v}^{comp} \tag{4.136}$$

where v^{reb} and v^{comp} are the volumes of the rebound and compression chambers. The role of the accumulator is to compensate for the variation of v^{tot}.



Rod	q^{piston}	q^{base}
$\uparrow$	$\dot{v}^{reb} < 0$	$\dot{v}^{tot} > 0$
$\downarrow$	$\dot{v}^{reb} > 0$	$\dot{v}^{tot} < 0$

Rod	q^{piston}	q^{base}	q^{CVSA}
$\uparrow$	0	$\dot{v}^{comp} > 0$	$\dot{v}^{reb} < 0$
$\downarrow$	$-\dot{v}^{comp} > 0$	0	$-\dot{v}^{tot} > 0$

Figure 4.25: Passive and semi-active damper. Quasi-static valve flows during compression and extension sequences. The proportions on the drawings are not realistic and, for information, typical values for the chamber diameter and the rod diameter are 32 mm and 22 mm, respectively.

In the active shock absorber, the passive valves are replaced by check valves, and a current controlled CVSA valve[4] connects the extreme chambers. When the rod moves up, the piston check-valve closes and the oil moves out of the rebound chamber through the CVSA valve. When the rod moves down, the base check-valve closes and the oil flows to the accumulator through the CVSA valve. Therefore, both motions are affected by the controlled restriction at the CVSA valve.

The flow variations reported in Figure 4.25 are related with a quasi-static motion, where the pressures are in equilibrium at every instant, the fluid is not compressible, and the accumulator is ideal. Actually, the dynamic behavior is more complex, and a nonlinear model has been calibrated by the manufacturer of the shock-absorber. This model can be presented in nonlinear state-space format,

[4]CVSA: continually variable semi-active.

defining the inputs, states and output:

$$\mathbf{u}^{(damp)} = [l^r \quad \dot{l}^r \quad i^v]^T \qquad \mathbf{x}^{(damp)} = [p^{reb} \quad p^{comp}]^T \qquad \mathbf{y}^{(damp)} = [g^a]^T \quad (4.137)$$

with l^r, the rattle extension, i^v, the electrical current in the CVSA valve, p^{reb} and p^{comp} the pressures in the rebound and compression chambers, and g^a the force exerted by the damper. The state equations are:

$$\begin{aligned} \dot{\mathbf{x}}^{(damp)} &= \mathbf{f}^{s(damp)}(\mathbf{u}^{(damp)}, \mathbf{x}^{(damp)}) \\ \mathbf{y}^{(damp)} &= \mathbf{f}^{o(damp)}(\mathbf{u}^{(damp)}, \mathbf{x}^{(damp)}) \end{aligned} \quad (4.138)$$

The functions $\mathbf{f}^{s(damp)}$ and $\mathbf{f}^{o(damp)}$ are given by the manufacturer as C-functions, which are linked with Oofelie and called in a user-element inheriting from the class `ContinuousSystem` (see Figure 4.9). The sensitivities of those functions are obtained using a finite difference procedure.

Controller model

The control law of the active suspension has been developed by Lauwerys, Swevers, and Sas [LSS04]. As illustrated in Figure 4.26, it consists of three stages: a feedback linearization (inverse actuator models), a transformation into modal space (coupling and decoupling operations), and a linear integral control.

The car and the dampers are represented as a black box, whose inputs are the four CVSA electrical currents i^v, and whose outputs are the rattle velocities $v^r = \dot{l}^r$, and the accelerations measured at the four corners of the car a^b (see Figure 4.23).

The feedback linearization technique seeks for virtual inputs which have the property to influence the outputs in a linear way. If one accepts that the non-linearity of the mechanism is weak, the main source of nonlinearity lies in the actuator. According to Lauwerys *et al.*, an efficient feedback linearization law is obtained by inversion of a simplified quasi-static model of the actuators. From equation (4.138), it is possible to formulate the quasi-static damper force with respect to the CVSA electrical current, and the rattle velocity:

$$g^a = g^a(i^v, v^r) \quad (4.139)$$

This relation can be inverted for $v^r \neq 0$:

$$i^v = (g^a)^{-1}\,(g^{virt}, v^r) \quad (4.140)$$

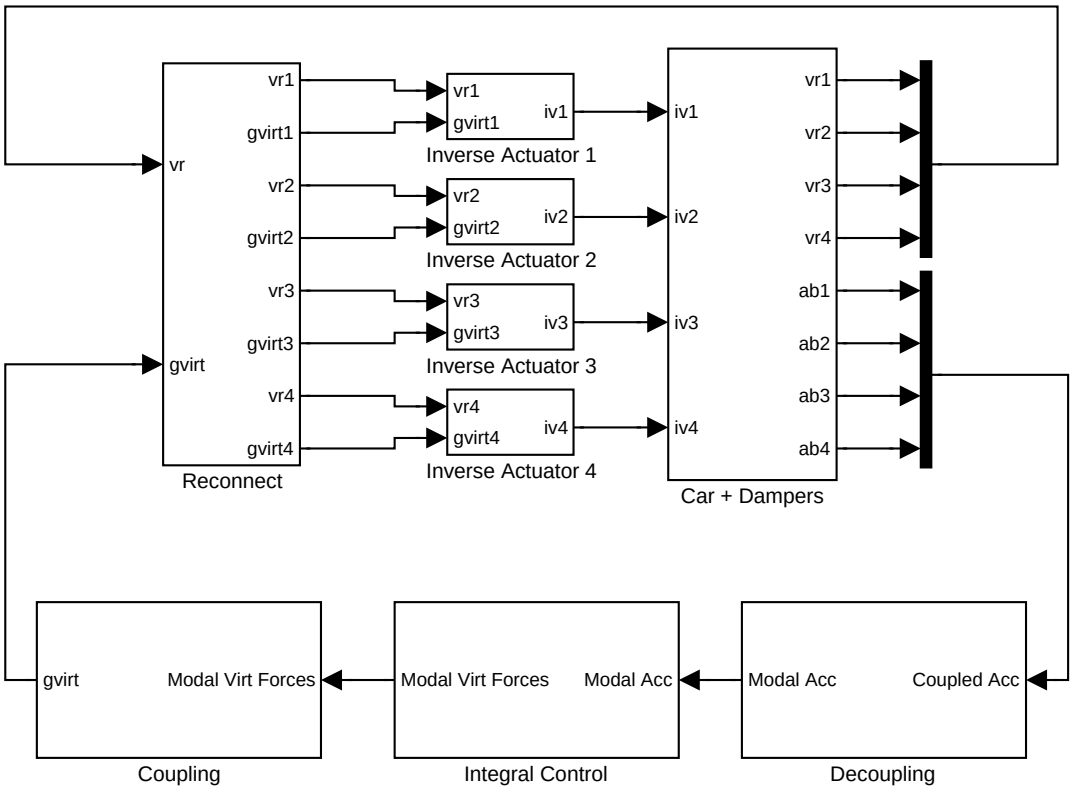


Figure 4.26: Controller of the semi-active suspension. ”vr” stands for the rattle velocities, ”ab” for the car-body accelerations, and ”iv” for the valve currents.

where g^{virt} is the new virtual input, which can be interpreted as a virtual damper force. For $v^r = 0$, the controllability is defective, and the singularity of $(g^a)^{-1}$ is avoided thanks to a regularization strategy. If this inverse model is able to cancel the nonlinearity of the actuator, the virtual input g^{virt} is actually proportional to the damper force. Therefore, besides the advantage of a good linearity between g^{virt} and the output, a force control strategy can be established on this basis.

Since the motion of the car simultaneously involves the forces applied on the four wheels, the definition of the virtual control forces g^{virt} for the four shock-absorbers is a linear but coupled multi-input/multi-output problem. This problem can be simplified by a transformation into a modal space defined by the heave (pumping), roll and pitch of the car-body. In this modal space, the system is represented by three uncoupled single-input/single-output subsystems, for which three independent integral controllers are designed.

The several stages of the controller are easily described using the block diagram language. The inverse actuator model is implemented as a specific element, which directly invokes the C-function implemented in the actual controller, and the sensitivities are obtained by analytical differentiation. All other blocks are modeled using the element library mentioned in Figure 4.9.

Simulation results

The simulation of a lane change maneuver has been realized, and the target trajectory, represented in Figure 4.27, corresponds to a standard qualification test. The car has a 10 m/s initial velocity, and a driver applies an open-loop steering command, without any real-time correction (blind driver assumption). The motor and the brakes do not produce any torque on the wheels. The time-step for the simulation is 0.01 s, and the algorithmic parameters are $\alpha_f = 0.05$ and $\alpha_m = 0$.

Figure 4.28 illustrates the motion of the slider-crank mechanism actuated by the driver, and the horizontal trajectory of the car.

Figure 4.29 represents the global motion of the car-body. The yaw, pitch and roll angles of the car-body are plotted as well as the radius of gyration and the vertical displacements. The radius of gyration becomes infinite when the car follows a straight line.

The dynamic behavior of the semi-active shock absorbers is illustrated in Figure 4.30. It is interesting to observe the pressures in the rebound and compression

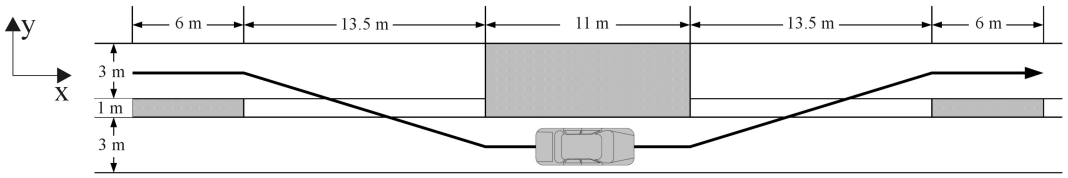


Figure 4.27: Lane change maneuver. The dimensions presented here correspond to a standard qualification test.

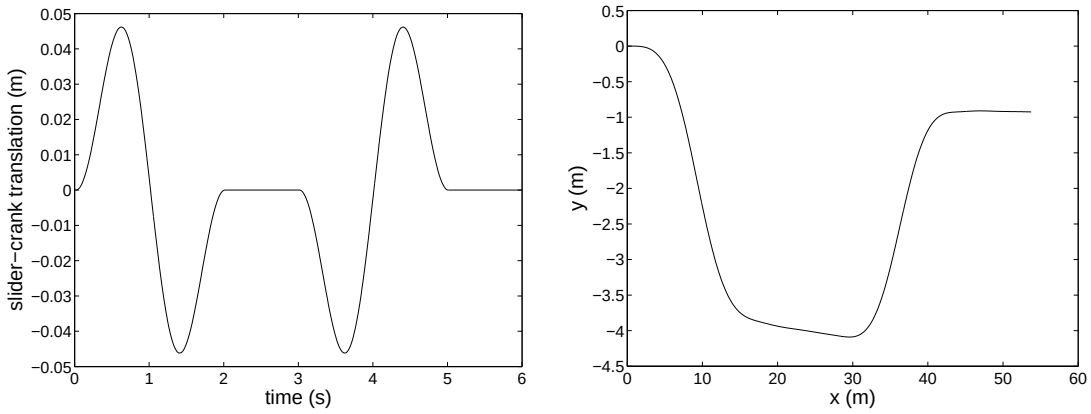


Figure 4.28: Car - steering command (slider-crank displacement) and horizontal trajectory of the car-body.

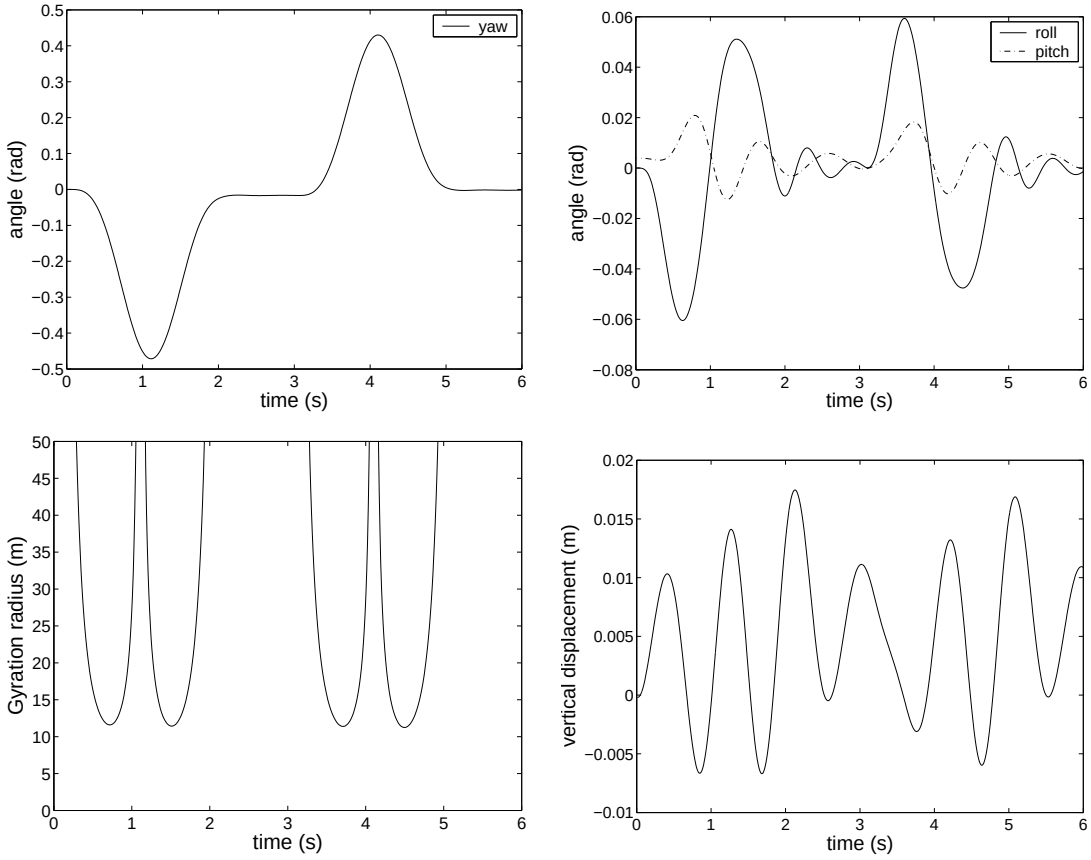


Figure 4.29: Car-body - angle and vertical displacements.

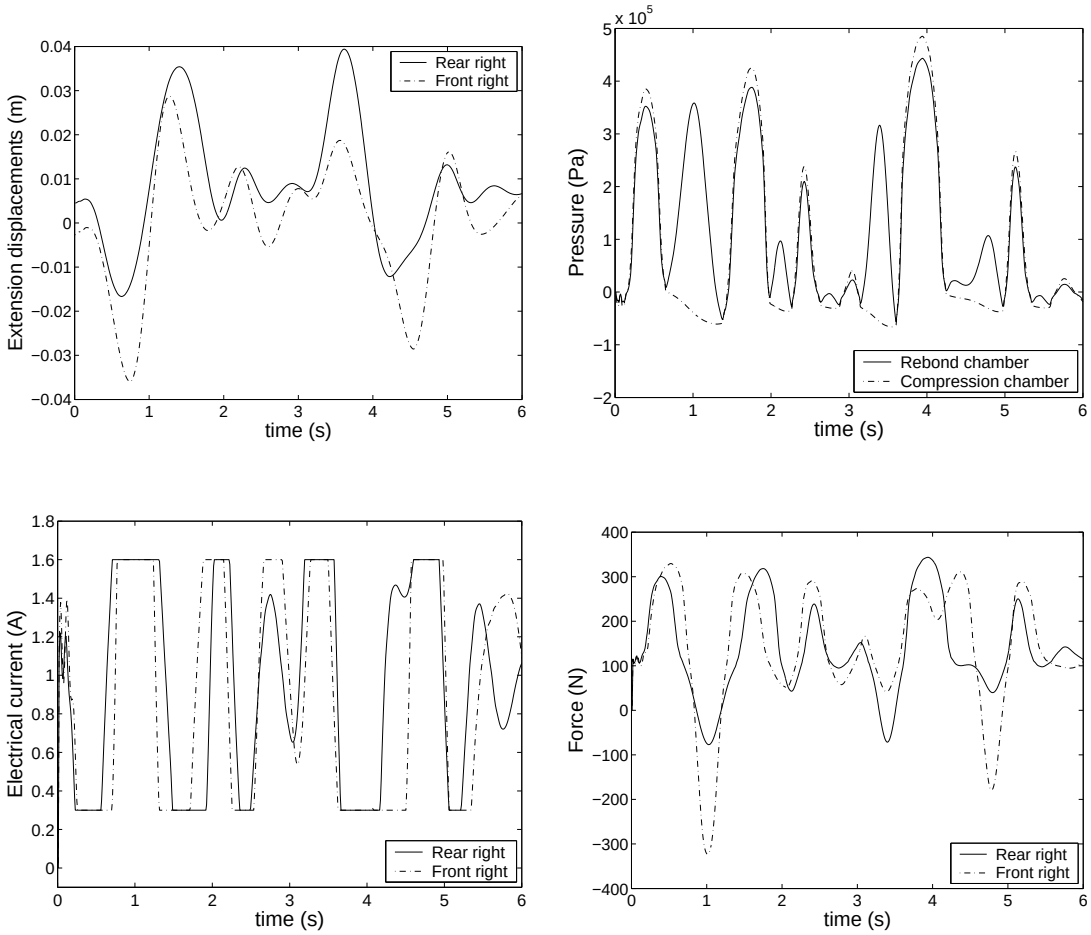


Figure 4.30: Car semi-active dampers - extension, hydraulic pressure (rear right), electrical current and damper force.

chambers, depending on the sign of the rattle velocity: during the compression, the valve between those chambers is open, and the pressure in the compression chamber is slightly higher, but during the extension, the valve is blocked, and the pressure in the rebound chamber can be much higher than in the compression chamber (see Figure 4.25). The saturation effect dominates the behavior of the electrical current in the CVSA valves. The static contribution of the forces produced by the actuators is not zero, due to non-equilibrated pressures in the different chambers when the piston rod is at rest.

In this application, the integrated simulation method was able to predict the behavior of a complex mechatronic system. The model consists of a full model of the car and its suspension mechanisms, a nonlinear dynamic model of the hy-

draulic actuator, and the nonlinear control algorithm. The modularity of the formulation leads to a systematic and convenient description of the mechatronic model. In particular, the block diagram language allows a direct definition of the controller model using the algorithm implemented in real-time, and vice versa. The simulation is carried out within a reasonable computational time (several minutes on a desk computer). From our experience with the simulation of purely mechanical systems using the Finite Element method, we believe that this performance could benefit from implementation optimization and from the development of an adaptive time-stepping strategy.

4.6 Summary and concluding remarks

In this chapter, well-defined concepts from multibody dynamics, structural dynamics and system theory have been integrated in a unified formulation for the simulation of mechatronic systems. In particular, our personal theoretical contributions concern:

- the adaptation of the block diagram concept for the modular representation of first-order state equations within a Finite Element formulation,

- the extension of the generalized-α method for the strongly coupled simulation of a mechatronic system represented by a mechanical Finite Element model and a block diagram model, with stability and convergence analyses,

- the development of a specific methodology to deal with possible discontinuous effects in some subsystems (*e.g.* sampled systems).

Three examples with increasing complexity have been successfully treated with this methodology: a four-bar mechanism, a Scara-robot, and a car semi-active suspension. All these examples involve rigid-body mechanisms. A fourth mechatronic system involving a flexible mechanism will be analyzed in chapter 6.

This systematic approach yields reliable results and it relieves the user from intricate, time-consuming and dubious implementation of user-elements for the dynamic description of a control system.

V

Nonlinear Model Reduction in Flexible Multibody Dynamics

In the state-of-the-art, section 2.4, new modeling concepts appeared to be necessary when designing a controller for a flexible multibody system with parallel topology. Indeed, a Finite Element model or an assumed-mode model are represented by a relatively high-order set of nonlinear DAEs, which is not convenient for this design problem. In this chapter, an original model reduction method is proposed leading to a suitable compromise between accuracy, order, structure, computational efficiency, systematic formulation and portability. The reduced model is obtained from an initial nonlinear Finite Element model, but since the theory assumes linear elasticity, a large part of the developments would still be valid if another initial formulation had been selected. The Finite Element method is preferred because of its systematic implementation, especially valuable when dealing with complex mechanisms.

Before starting the detailed presentation of the reduction method, an overview is first presented in order to draw the reader's attention to the key steps and features of the approach.

5.1 Introduction

The reduction method is an extension of the component-mode technique established in structural dynamics, which accounts for the nonlinear kinematics of the mechanism. Basically, the reduction procedure proceeds in two steps:

- an order reduction of the initial Finite Element model, which can be realized locally for any given configuration,

- an approximation of the reduced-order model in the configuration space.

Let us further develop both steps.

The order reduction comes from a reduced kinematic description with modal coordinates associated with the motion of the *whole* mechanism, including all bodies and joints. The mode shapes are therefore *configuration-dependent*: for every configuration, their definition according to a component-mode synthesis is based on the linearized dynamic equations. Since those modes are not associated with a particular component of the mechanism, and since they are defined locally for small motions around a configuration, we refer to them as *local modes* rather than component-modes. The local mode shapes characterize a local coordinate transformation between the initial Finite Element coordinates $\mathbf{q}$ and the modal coordinates $\boldsymbol{\eta}$. The reduced-order model is obtained after introduction of this coordinate transformation into the initial model.

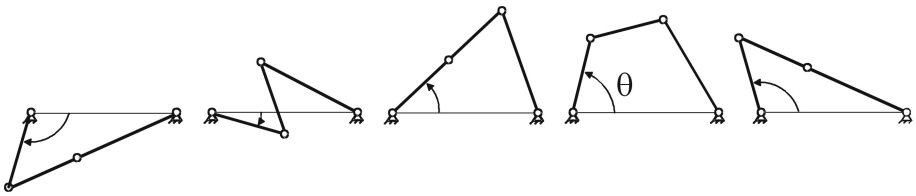
This local reduction procedure, which should be repeated for every configuration change, is computationally expensive. A more suitable model can be constructed by approximation of the reduced-order model in the configuration space [BDG04c].

An implementation of this approach is illustrated in Figure 5.1: a number of reference kinematic configurations $\boldsymbol{\theta}^{(i)}$, $(i = 1, ..., r)$ are selected in the configuration space, the Finite Element model is built and reduced for each of them, afterwards an approximation algorithm leads to a closed-form model.

The validity and the consistency of this approach is questionable for two reasons:

- the modal coordinates defined in one configuration may not be naturally associated with the modal coordinates in another configuration, so that non-physical discontinuous transitions may appear in the reduced model,

- the modal parameterization is intrinsically nonlinear, and the standard model reduction procedure defined in linear structural dynamics has to be reconsidered.



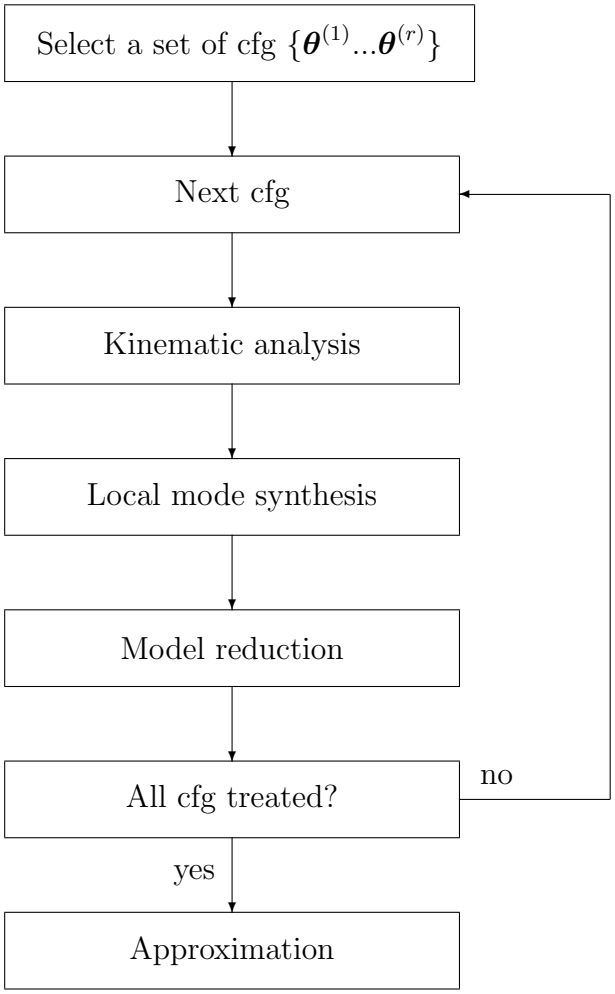


Figure 5.1: Simple approach for the model reduction of a flexible mechanism, and illustration with a four-bar mechanism.

In order to guarantee the consistency of the model, we assess the reduction method on a well-defined nonlinear reduced parameterization, valid in the whole configuration space, and consistent with the local modes. This parameterization relies on separated descriptions of the large amplitude rigid motion, and of the small amplitude flexible motion [BDG04a]. Since it is related with mode shapes of the global mechanism, defined in the global configuration space, it is denoted the *Global Modal Parameterization*. The formal definition of this concept is an important contribution of this dissertation.

A second contribution is associated with the numerical reduction procedure, which accounts for the nonlinear nature of the Global Modal Parameterization. Actually, it will be demonstrated that the curvature of the coordinate transformation leads to a contribution to the reduced inertia forces.

The development of a piecewise approximation strategy in the configuration space is the third original contribution of this chapter. The method follows an adaptive configuration space inspection algorithm, which minimizes the computational resources to satisfy a user-defined tolerance on the approximation error.

The reduced model satisfies the following specifications:

- accuracy in the bandwidth of the actuators and within the limits of the workspace,
- low-order and free from kinematic constraints, which follows from the Global Modal Parameterization,
- physical interpretation of the modal coordinates in terms of rigid and flexible modes, and interpretation of the model parameters in terms of mass matrix, stiffness matrix, and gyroscopic tensor,
- available in closed-form, involving inexpensive computations,
- formulated systematically from high-level information,
- defined in a simple standard format, easily exported to other software for control design.

This model can be exploited for different purposes, such as:

- the online implementation of a feedback controller,

- the offline optimization of a control policy,

- the construction of a more structured model, in order to fit well-established control theories.

After this preliminary description, section 5.2 presents component-mode techniques developed in linear structural dynamics. Section 5.3 is devoted to the characterization of the nonlinear kinematics of parallel mechanisms. The Global Modal Parameterization and the local reduction procedure are respectively described in sections 5.4 and 5.5. The approximation strategy is presented in section 5.6. After a summary of the overall algorithm in section 5.7, two applications are treated in section 5.8: a flexible four-bar mechanism, and a rigid parallel-kinematic machine-tool (a third example is left for chapter 6). The chapter ends with a few concluding remarks.

5.2 Linear component-mode synthesis

A linear mechanical system is represented by the equations of motion:

$$\mathbf{M}\,\ddot{\mathbf{q}} + \mathbf{K}\,\mathbf{q} = \mathbf{g} \tag{5.1}$$

with $\mathbf{q}$, the $n \times 1$ vector of generalized coordinates, $\mathbf{M}$, the mass matrix, $\mathbf{K}$, the stiffness matrix, and $\mathbf{g}$, the vector of applied force. A reduced-order model offers obvious advantages associated with the simplified treatments required for its analysis.

In structural dynamics [GR97], the development of model reduction techniques has also been motivated by substructuring applications, where the analysis of a complex dynamic structure exploits a decomposition into several simpler parts. A reduced-order model is then constructed to represent concisely the dynamics of each substructure, and it will later be used to reconstruct the global model. This approach allows a detailed modeling of components with complex geometry, while keeping a relatively simple global dynamic model. It also leads to an increased modularity and reusability in the modeling process. In our context, the mechanical system can be seen as a "substructure" of the mechatronic system, for which a reduced-order model is desirable.

5.2.1 Principle of a linear reduction method

A reduction method relies on the construction of a change of variables from initial coordinates to modal coordinates according to

$$\overline{\mathbf{q}} = \mathbf{\Psi}\ \boldsymbol{\eta} \tag{5.2}$$

where $\boldsymbol{\eta}$ is the $\overline{n} \times 1$ vector of modal coordinates, and $\mathbf{\Psi}$ is the $n \times \overline{n}$ matrix of component-modes. The reduction comes from

$$\overline{n} < n \tag{5.3}$$

The variations of $\overline{\mathbf{q}}$ are thus restricted to the subspace spanned by the component-modes, and the accuracy of the reduced model depends on the ability of the component-modes to describe the actual motion of the system.

In order to formulate the reduced equations of motion, the coordinate transformation is introduced into the expressions of the kinetic and potential energies, and of the virtual work of the external forces. Initially, we have:

$$\mathcal{K} = \frac{1}{2}\dot{\mathbf{q}}^T\ \mathbf{M}\ \dot{\mathbf{q}} \tag{5.4}$$

$$\mathcal{V} = \frac{1}{2}\mathbf{q}^T\ \mathbf{K}\ \mathbf{q} \tag{5.5}$$

$$\delta\mathcal{W} = \mathbf{g}^T\ \delta\mathbf{q} \tag{5.6}$$

and the transformation into modal coordinates leads to

$$\mathcal{K} = \frac{1}{2}\dot{\boldsymbol{\eta}}^T\ \overline{\mathbf{M}}\ \dot{\boldsymbol{\eta}} \tag{5.7}$$

$$\mathcal{V} = \frac{1}{2}\boldsymbol{\eta}^T\ \overline{\mathbf{K}}\ \boldsymbol{\eta} \tag{5.8}$$

$$\delta\mathcal{W} = \overline{\mathbf{g}}^T\ \delta\boldsymbol{\eta} \tag{5.9}$$

with the reduced mass matrix, stiffness matrix, and equivalent force vector:

$$\overline{\mathbf{M}} = \mathbf{\Psi}^T\,\mathbf{M}\,\mathbf{\Psi}, \qquad \overline{\mathbf{K}} = \mathbf{\Psi}^T\,\mathbf{K}\,\mathbf{\Psi}, \qquad \overline{\mathbf{g}} = \mathbf{\Psi}^T\,\mathbf{g} \tag{5.10}$$

The equations of motion of the reduced system follow:

$$\overline{\mathbf{M}}\ \ddot{\boldsymbol{\eta}} + \overline{\mathbf{K}}\ \boldsymbol{\eta} = \overline{\mathbf{g}} \tag{5.11}$$

A similar procedure can be applied for a system with m linear constraints:

$$\mathbf{\Phi_q}\ \mathbf{q} = \mathbf{0} \tag{5.12}$$

Following an augmented Lagrangian approach, an $m \times 1$ vector of Lagrange multipliers $\boldsymbol{\lambda}$ is introduced, and the vector of augmented generalized coordinates is defined:

$$\mathbf{u} = \begin{bmatrix} \mathbf{q} \\ \boldsymbol{\lambda} \end{bmatrix} \tag{5.13}$$

as well as the matrices and vector:

$$\mathbf{M^{uu}} = \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \qquad \mathbf{K^{uu}} = \begin{bmatrix} \mathbf{K} + p\,\boldsymbol{\Phi}_\mathbf{q}^T \boldsymbol{\Phi}_\mathbf{q} & k\,\boldsymbol{\Phi}_\mathbf{q}^T \\ k\,\boldsymbol{\Phi}_\mathbf{q} & \mathbf{0} \end{bmatrix}, \qquad \mathbf{g^u} = \begin{bmatrix} \mathbf{g} \\ \mathbf{0} \end{bmatrix} \tag{5.14}$$

where k and p are the scaling and penalty factors. The constrained equations of motion are restated:

$$\mathbf{M^{uu}}\,\ddot{\mathbf{u}} + \mathbf{K^{uu}}\,\mathbf{u} = \mathbf{g^u} \tag{5.15}$$

and the component-mode technique can be applied as in the unconstrained case:

$$\overline{\mathbf{u}} = \boldsymbol{\Psi}^{\mathbf{u}\boldsymbol{\eta}}\,\boldsymbol{\eta} \tag{5.16}$$

5.2.2 Component-mode selection

The objective of the component-mode synthesis is to define the mode shapes $\boldsymbol{\Psi}$, leading to a minimized loss in accuracy. Let us consider a partitioning of the generalized coordinates:

$$\mathbf{u} = \begin{bmatrix} \mathbf{q}^r \\ \mathbf{q}^g \\ \mathbf{u}^i \end{bmatrix} \tag{5.17}$$

where $\mathbf{q}^r$ are s rigid dofs strictly able to characterize the rigid modes, $\mathbf{q}^g$ are n^g constraint dofs where external loads are applied[1], and $\mathbf{u}^i$ are the n^i remaining internal dofs which are not loaded and are condensed by the reduction procedure. The set of internal dofs includes the Lagrange multipliers.

On this basis, Hurty [Hur65] proposed three subsets of modes: the rigid-body modes, the constraint modes, and the internal modes. This choice leads to an exact representation of the static response of the structure, which is critical for the consistent assembly of the reduced model with other substructures. In a

[1] In a substructuring framework, $\mathbf{q}^g$ also includes interface dofs between substructures, subject to internal forces.

mechatronic framework, the static response of a structure should also be accurately represented since it influences the zeros of the mechanical transfer functions, as discussed by Preumont [Pre97]. This technique was improved by Craig and Bampton [CB68] who suggested to treat all boundary degrees of freedom ($\mathbf{q}^r$ and $\mathbf{q}^g$) alike. It can be demonstrated that the modes proposed by Hurty and by Craig-Bampton span the same subspace, so that the accuracy of the reduced model is equivalent for both methods. The Craig-Bampton method offers several advantages, such as a simplified partitioning, a greater efficiency, and the explicit conservation of all boundary dofs in the reduced model. This is probably the reason why it is so popular and available in most commercial Finite Element software. However, the rigid modes do not appear explicitly among the boundary modes, which is a precluding drawback for us: our nonlinear reduction method entirely relies on the separation between the rigid and the flexible motion.

For this reason, the component-modes defined by Hurty [Hur65] are exploited here, but this choice is not restrictive. Many other component-modes have been defined in the literature, see Craig [Cra87] and De Fonseca [DF00] for detailed reviews, and our nonlinear reduction method could involve any of them, provided that the rigid-body modes are isolated from the flexible modes.

The s rigid-body modes $\mathbf{\Psi}^{\mathbf{u}\boldsymbol{\theta}}$ are defined with respect to the rigid dofs $\mathbf{q}^r$, and they satisfy the equilibrium of internal forces:

$$\begin{bmatrix} \mathbf{K}^{rr} & \mathbf{K}^{rg} & \mathbf{K}^{ri} \\ \mathbf{K}^{gr} & \mathbf{K}^{gg} & \mathbf{K}^{gi} \\ \mathbf{K}^{ir} & \mathbf{K}^{ig} & \mathbf{K}^{ii} \end{bmatrix} \begin{bmatrix} \mathbf{I}^{r\boldsymbol{\theta}} \\ \mathbf{\Psi}^{g\boldsymbol{\theta}} \\ \mathbf{\Psi}^{i\boldsymbol{\theta}} \end{bmatrix} = \mathbf{0}, \qquad \mathbf{\Psi}^{\mathbf{u}\boldsymbol{\theta}} = \begin{bmatrix} \mathbf{I}^{r\boldsymbol{\theta}} \\ \mathbf{\Psi}^{g\boldsymbol{\theta}} \\ \mathbf{\Psi}^{i\boldsymbol{\theta}} \end{bmatrix} \tag{5.18}$$

$\mathbf{I}^{r\boldsymbol{\theta}}$ is the $s \times s$ identity matrix.

The n^g constraint modes $\mathbf{\Psi}^{\mathbf{u}\gamma}$ are the static deformations obtained when the rigid dofs $\mathbf{q}^r$ are fixed and unit displacements are imposed on the constrained dofs $\mathbf{q}^g$:

$$\begin{bmatrix} \mathbf{K}^{gg} & \mathbf{K}^{gi} \\ \mathbf{K}^{ig} & \mathbf{K}^{ii} \end{bmatrix} \begin{bmatrix} \mathbf{I}^{g\gamma} \\ \mathbf{\Psi}^{i\gamma} \end{bmatrix} = \begin{bmatrix} \mathbf{g}^g \\ \mathbf{0} \end{bmatrix}, \qquad \mathbf{\Psi}^{\mathbf{u}\gamma} = \begin{bmatrix} \mathbf{0}^{r\gamma} \\ \mathbf{I}^{g\gamma} \\ \mathbf{\Psi}^{i\gamma} \end{bmatrix} \tag{5.19}$$

$\mathbf{I}^{g\gamma}$ is the $n^g \times n^g$ identity matrix.

The n^ι internal modes $\mathbf{\Psi}^{\mathbf{u}\iota}$ are a few eigenmodes when rigid and constraint

dofs are fixed ($n^\iota < n^i$).

$$\left(\mathbf{K}^{ii} - \omega^2\, \mathbf{M}^{ii}\right)\ \mathbf{\Psi}^{i\iota} = \mathbf{0}, \qquad \mathbf{\Psi}^{\mathbf{u}\iota} = \begin{bmatrix} \mathbf{0}^{r\iota} \\ \mathbf{0}^{g\iota} \\ \mathbf{\Psi}^{i\iota} \end{bmatrix} \tag{5.20}$$

The full transformation matrix follows:

$$\begin{bmatrix} \mathbf{q}^r \\ \mathbf{q}^g \\ \mathbf{u}^i \end{bmatrix} = \begin{bmatrix} \mathbf{\Psi}^{\mathbf{u}\boldsymbol{\theta}} & \mathbf{\Psi}^{\mathbf{u}\gamma} & \mathbf{\Psi}^{\mathbf{u}\iota} \end{bmatrix} \begin{bmatrix} \boldsymbol{\theta} \\ \boldsymbol{\eta}^\gamma \\ \boldsymbol{\eta}^\iota \end{bmatrix} = \begin{bmatrix} \mathbf{I}^{r\boldsymbol{\theta}} & \mathbf{0} & \mathbf{0} \\ \mathbf{\Psi}^{g\boldsymbol{\theta}} & \mathbf{I}^{g\gamma} & \mathbf{0} \\ \mathbf{\Psi}^{i\boldsymbol{\theta}} & \mathbf{\Psi}^{i\gamma} & \mathbf{\Psi}^{i\iota} \end{bmatrix} \begin{bmatrix} \boldsymbol{\theta} \\ \boldsymbol{\eta}^\gamma \\ \boldsymbol{\eta}^\iota \end{bmatrix} \tag{5.21}$$

where $\boldsymbol{\theta}$, $\boldsymbol{\eta}^\gamma$ and $\boldsymbol{\eta}^\iota$ denote the modal amplitudes. By construction, the amplitudes of the rigid modes $\boldsymbol{\theta}$ are identified with the rigid dofs $\mathbf{q}^r$:

$$\mathbf{q}^r = \boldsymbol{\theta} \tag{5.22}$$

This property does not hold for the constraint modes: $\mathbf{q}^g \neq \boldsymbol{\eta}^\gamma$. If the modal masses of the internal modes are normalized, the reduced matrices and forces have the following structure:

$$\overline{\mathbf{K}} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \overline{\mathbf{K}}^{\gamma\gamma} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{\Omega}^2 \end{bmatrix},\ \overline{\mathbf{M}} = \begin{bmatrix} \overline{\mathbf{M}}^{\boldsymbol{\theta}\boldsymbol{\theta}} & \overline{\mathbf{M}}^{\boldsymbol{\theta}\gamma} & \overline{\mathbf{M}}^{\boldsymbol{\theta}\iota} \\ \overline{\mathbf{M}}^{\gamma\boldsymbol{\theta}} & \overline{\mathbf{M}}^{\gamma\gamma} & \overline{\mathbf{M}}^{\gamma\iota} \\ \overline{\mathbf{M}}^{\iota\boldsymbol{\theta}} & \overline{\mathbf{M}}^{\iota\gamma} & \mathbf{I}^{\iota\iota} \end{bmatrix},\ \begin{bmatrix} \overline{\mathbf{g}}^{\boldsymbol{\theta}} \\ \overline{\mathbf{g}}^{\gamma} \\ \overline{\mathbf{g}}^{\iota} \end{bmatrix} = \begin{bmatrix} \mathbf{g}^r + \mathbf{\Psi}^{g\boldsymbol{\theta}T}\, \mathbf{g}^g \\ \mathbf{g}^g \\ \mathbf{0} \end{bmatrix} \tag{5.23}$$

where $\mathbf{\Omega} = \mathrm{diag}(\omega_i)$ is the diagonal matrix of internal eigenvalues.

Géradin and Rixen [GR97] interpreted a class of component-mode methods as a truncation in the modal expansion of the mechanical impedance. In this sense, the component-modes suggested by Hurty and Craig-Bampton are optimal.

In multibody dynamics, the linear component-mode technique is usually exploited for the compact kinematic description of an *isolated flexible body* with respect to a floating frame of reference, see section 2.1.2. In order to obtain a more drastic reduction, we apply the modal parameterization to the *whole mechanism*, which is seen as a "component" of the mechatronic system. Therefore, the concept of component-mode is replaced by the concept of local mode defined around a configuration. The description of the configuration of a mechanism with a suitable parameterization is addressed in the next section.

5.3 Parameterization of the rigid kinematics

The parameterization of the motion of a mechanism has been considered in section 3.1. Several sets of parameters were described, such as the minimal coordinates, the relative coordinates, the Cartesian coordinates, the absolute coordinates, and the mixed coordinates. The strong advantage of minimal coordinates comes from their independence, but we mentioned that a minimal parameterization is only possible in a restricted part of the configuration space. Thus, dependent coordinates were preferred for the formulation of general analysis methods.

However, for control design, we will demonstrate that the minimal coordinates can be exploited with great benefits. Our objective is to formulate the dynamics of flexible mechanisms in terms of minimal coordinates, *i.e.* free from kinematic constraints.

Under the assumption of small deformations, the flexible motion can be described in the neighborhood of the motion of the undeformed mechanism. A first problem is then to parameterize the rigid kinematics associated with the flexible mechanism, using independent coordinates. This section specifically addresses this problem.

As for linear reduction methods, the approach is based on a coordinate transformation. Here, we rely on an initial parameterization in terms of dependent coordinates $\mathbf{q}$ (*e.g.* Finite Element coordinates), and we seek for a transformation into independent coordinates $\boldsymbol{\theta}$. Therefore, two problems are encountered:

- the definition of the independent parameters $\boldsymbol{\theta}$,
- the characterization of the nonlinear coordinate transformation $\mathbf{q} = \boldsymbol{\rho}(\boldsymbol{\theta})$ and its Jacobian $\boldsymbol{\rho}_{\boldsymbol{\theta}}$.

The formulation of the second problem is substantially different whether the initial parameters $\mathbf{q}$ account or not for the deformations. For this reason, we successively consider the case of rigid mechanisms and the case of flexible mechanisms.

5.3.1 Rigid mechanisms

The motion of a rigid mechanism can be described with a vector of *dependent coordinates* $\mathbf{q} \in \mathcal{R}^n$, satisfying m holonomic constraint equations:

$$\boldsymbol{\Phi}(\mathbf{q}) = \mathbf{0} \tag{5.24}$$

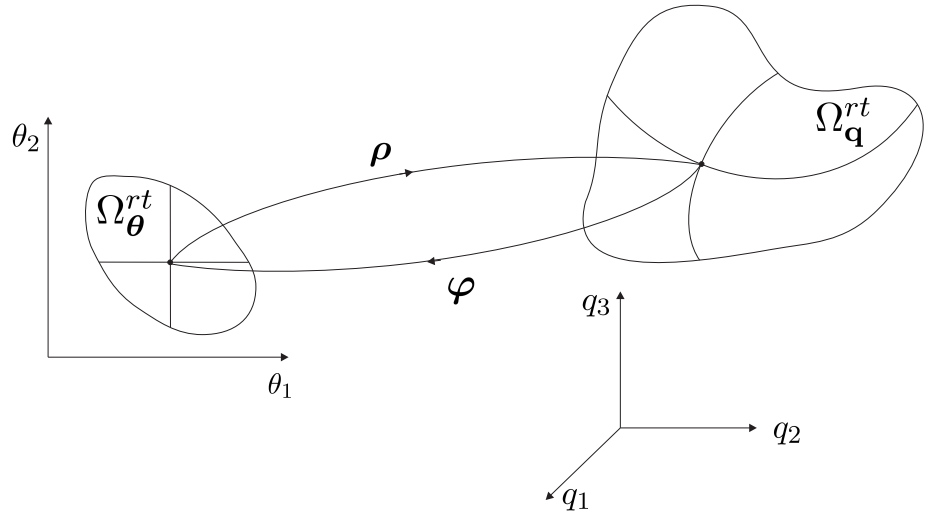


Figure 5.2: Coordinate transformation between dependent and independent parameters.

The *configuration space* $\Omega^{rt}_{\mathbf{q}}$ is defined as the set of kinematically admissible configurations which satisfy the kinematic constraints:

$$\Omega^{rt}_{\mathbf{q}} = \{\mathbf{q} \in \mathcal{R}^n \mid \boldsymbol{\Phi}(\mathbf{q}) = \mathbf{0}\} \tag{5.25}$$

The superscript r stands for "rigid", and t for the "total" configuration space. Assuming independent constraints, the number of kinematic modes s satisfies:

$$s = n - m \tag{5.26}$$

We seek for a $s \times 1$ vector of minimal parameters $\boldsymbol{\theta}$, which are related to $\mathbf{q}$ by a coordinate transformation, represented by an invertible mapping $\boldsymbol{\rho} = \boldsymbol{\varphi}^{-1}$:

$$\boldsymbol{\rho} : \Omega^{rt}_{\boldsymbol{\theta}} \to \Omega^{rt}_{\mathbf{q}}, \qquad \boldsymbol{\theta} \mapsto \mathbf{q} = \boldsymbol{\rho}(\boldsymbol{\theta}) \tag{5.27}$$

$$\boldsymbol{\varphi} : \Omega^{rt}_{\mathbf{q}} \to \Omega^{rt}_{\boldsymbol{\theta}}, \qquad \mathbf{q} \mapsto \boldsymbol{\theta} = \boldsymbol{\varphi}(\mathbf{q}) \tag{5.28}$$

where $\Omega^{rt}_{\boldsymbol{\theta}}$ is the set of possible variations of the parameters $\boldsymbol{\theta}$. We also refer to $\Omega^{rt}_{\boldsymbol{\theta}}$ as the configuration space, whenever no confusion is possible. Figure 5.2 illustrates those definitions.

In general, the existence of a global parameterization with independent coordinates is not possible in the total configuration space $\Omega^{rt}_{\mathbf{q}}$, and some restrictions are necessary to define a pragmatic solution.

We propose to define the minimal coordinates $\boldsymbol{\theta}$ as the *actuated dofs*, *i.e.* the dofs associated with the generalized forces exerted by the actuators. For instance, for a motorized hinge, the actuated dof is the angle between the connected links, whereas for a linear actuator, it is the relative distance between the connected bodies. As a consequence of this choice, the actuator dofs will appear explicitly in the reduced model, which is extremely valuable for the design of the control system. An implicit assumption is that the number of actuators is equal to s. This does not imply that the reduction method is only applicable to fully actuated mechanisms, it only means that in any other situation, the definition of the independent parameters is left to the user.

The actuated dofs $\boldsymbol{\theta}$ are usually relative coordinates, in contrast with the absolute coordinates $\mathbf{q}$ used in our Finite Element formulation. A systematic implementation of the relation $\boldsymbol{\theta} = \boldsymbol{\varphi}(\mathbf{q})$ can be defined using *mixed coordinates*. This means that the actuator coordinates $\boldsymbol{\theta}$ appear explicitly among the set of n mixed coordinates $\mathbf{q}$:

$$\mathbf{q} = \begin{bmatrix} \boldsymbol{\theta} \\ \mathbf{q}^* \end{bmatrix} \tag{5.29}$$

$\mathbf{q}^*$ are the $n-s$ non-actuated dofs. Since $m = n - s$, the number of non-actuated dofs equals the number of kinematic constraints of the mixed formulation.

The *mapping* $\boldsymbol{\varphi}$ is directly characterized:

$$\boldsymbol{\theta} = \boldsymbol{\varphi}(\mathbf{q}) = [\mathbf{I} \quad \mathbf{0}] \ \mathbf{q} \tag{5.30}$$

where $\mathbf{I}$ is the $s \times s$ identity matrix and $\mathbf{0}$ is the $s \times (n-s)$ null matrix.

The formulation of the *inverse mapping* $\boldsymbol{\rho}$ requires a deeper investigation. Mathematically, the n equations:

$$\begin{cases} \boldsymbol{\varphi}(\mathbf{q}) = \boldsymbol{\theta} \\ \boldsymbol{\Phi}(\mathbf{q}) = \mathbf{0} \end{cases} \tag{5.31}$$

can be interpreted as an implicit definition for $\mathbf{q} = \boldsymbol{\rho}(\boldsymbol{\theta})$. According to the implicit function theorem, the existence of $\boldsymbol{\rho}$ is guaranteed *locally*, if the Jacobian of the nonlinear system (5.31) is non-singular. In other words, the gradient $\boldsymbol{\varphi}_{\mathbf{q}}$, and the constraint gradient $\boldsymbol{\Phi}_{\mathbf{q}}$ should be linearly independent. Configurations for which this condition is not satisfied are either configuration space singularities or actuator singularities [PK99, ZBG01, LLL03]. At a *configuration space singularity*, the

constraint gradient $\mathbf{\Phi_q}$ becomes defective, and the number of kinematic modes increases suddenly. Those singularities are intrinsic properties of the configuration space, independent of the parameterization problem. In contrast, the location of an *actuator singularity* (or parameterization singularity) is conditioned by the choice of the actuated dofs. At an actuator singularity, the mechanism can move even though all actuators are blocked, which is usually undesirable from a control point of view.

The regularity of the Jacobian is not sufficient to guarantee that the actuator dofs $\boldsymbol{\theta}$ are able to parameterize the whole configuration space. A well-defined parameterization is *globally* one-to-one: for each given actuator configuration $\boldsymbol{\theta}$, there exists only one kinematically admissible configuration $\mathbf{q}$ (*i.e.* one solution to equations (5.31)). This condition is quite difficult to verify; however, in many practical cases, actuator singularities separate parts of the configuration space where $\boldsymbol{\varphi}$ is one-to-one, so that the actuator parameterization is valid in a subset $\Omega^r_{\mathbf{q}} \subset \Omega^{rt}_{\mathbf{q}}$ bounded by the singular configurations. All those concepts are illustrated in Figure 5.3.

In this restricted part of the configuration space, the inverse map $\boldsymbol{\rho}$ is well-defined:

$$\boldsymbol{\rho} : \Omega^r_{\boldsymbol{\theta}} \to \Omega^r_{\mathbf{q}}, \qquad \boldsymbol{\theta} \mapsto \mathbf{q} = \boldsymbol{\rho}(\boldsymbol{\theta}) \tag{5.32}$$

where $\Omega^r_{\boldsymbol{\theta}} \subset \Omega^{rt}_{\boldsymbol{\theta}} \subset \mathcal{R}^s$ is the set of possible variations of the actuated dofs $\boldsymbol{\theta}$, away from the singularities. For usual control applications, actuator singularities and configuration space singularities are carefully avoided, and the limitation of the analysis to $\Omega^r_{\mathbf{q}}$ is not restrictive.

Jacobian of the transformation

In the space of dependent coordinates $\mathbf{q}$, a relation exists between the constraint gradient and the Jacobian of $\boldsymbol{\rho}$. Indeed, we have:

$$\mathbf{\Phi}(\boldsymbol{\rho}(\boldsymbol{\theta})) = \mathbf{0}, \quad \forall\, \boldsymbol{\theta} \in \Omega^r_{\boldsymbol{\theta}} \tag{5.33}$$

so that, after differentiation:

$$\mathbf{\Phi_q}\, \boldsymbol{\rho}_{\boldsymbol{\theta}} = \mathbf{0} \tag{5.34}$$

$\boldsymbol{\rho}_{\boldsymbol{\theta}}$ is the Jacobian of the coordinate transformation, which naturally spans the tangent space of the configuration manifold $T_p\Omega^r_{\mathbf{q}}$ at point p. According to (5.34),

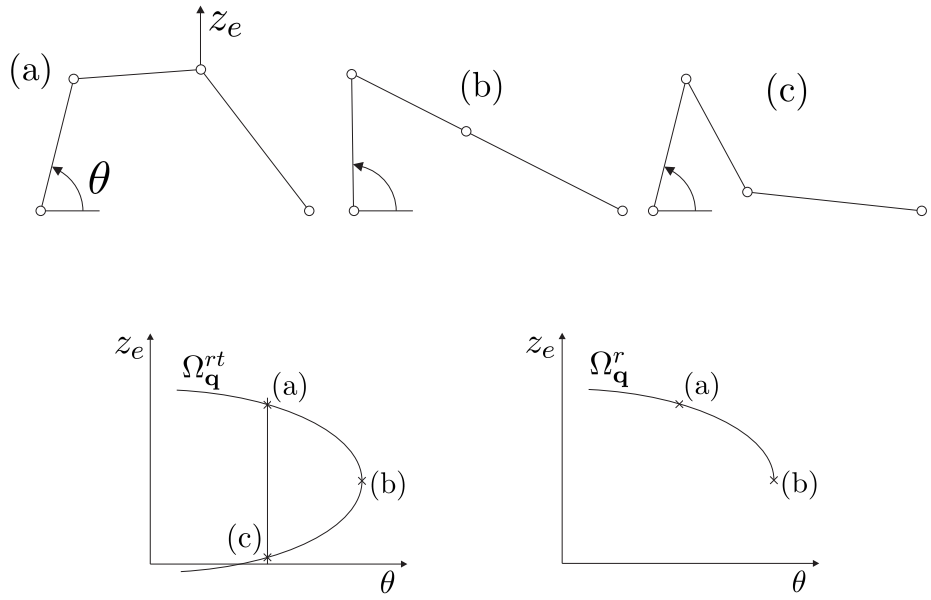


Figure 5.3: Rigid four bar mechanism, with an actuator at the bottom left hinge. The configuration space is a 1-dimensional manifold, whose projection in the plane of the coordinates z_e and θ is also represented. Configuration (b) is an actuator singularity; it separates two sub-domains of the configuration space for which $\mathbf{q} = \boldsymbol{\varphi}(\boldsymbol{\theta})$ is one-to-one. $\boldsymbol{\varphi}$ is not one-to-one in the total configuration space $\Omega^{rt}_{\mathbf{q}}$, since configurations (a) and (c) are possible for the same angle θ.

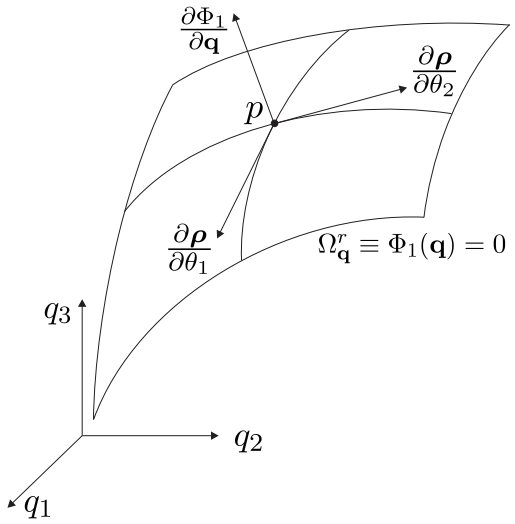


Figure 5.4: Tangent space and constraint gradient of a rigid mechanism.

$\boldsymbol{\rho}_{\boldsymbol{\theta}}$ is orthogonal to the columns of the constraint gradient $\boldsymbol{\Phi}^T_{\mathbf{q}}$, as represented in Figure 5.4.

To conclude this discussion, it is worth noticing that the definition of independent parameters $\boldsymbol{\theta}$ among the generalized coordinates $\mathbf{q}$ is formally equivalent to the *constraint elimination technique* described in section 3.2.1. Hence, for a rigid mechanism, the constraint elimination technique can be exploited for model reduction, as proposed in a previous work [BDG06]. In this dissertation, we focus on the more general case of flexible mechanisms; but the methodology can still be reinterpreted as a constraint elimination technique if the mechanism is actually rigid.

5.3.2 Flexible mechanisms

In the previous section, the analysis was based on a set of dependent parameters subject to kinematic constraints. For a flexible mechanism, additional constraints of non-deformation are required for the rigid kinematic analysis.

The n coordinates $\mathbf{q}$ still have to satisfy m kinematic constraints:

$$\boldsymbol{\Phi}(\mathbf{q}) = \mathbf{0} \tag{5.35}$$

The *flexible manifold* $\Omega^{ft}_{\mathbf{q}}$ is defined as the set of kinematically admissible config-

urations which satisfy the kinematic constraints:

$$\Omega_{\mathbf{q}}^{ft} = \{\mathbf{q} \in \mathcal{R}^n \mid \mathbf{\Phi}(\mathbf{q}) = \mathbf{0}\} \tag{5.36}$$

Moreover, the *configuration space* $\Omega_{\mathbf{q}}^{rt} \subset \Omega_{\mathbf{q}}^{ft}$ is defined as the set of *undeformed configurations*. The dimension[2] s of $\Omega_{\mathbf{q}}^{rt}$ is smaller than the dimension $n - m$ of $\Omega_{\mathbf{q}}^{ft}$:

$$s < n - m \tag{5.37}$$

As for rigid mechanisms, the actuated dofs $\boldsymbol{\theta}$ lead to a parameterization of a restricted part of the configuration space $\Omega_{\mathbf{q}}^{r} \subset \Omega_{\mathbf{q}}^{rt}$. According to the mixed coordinate formulation, those dofs are included among the generalized coordinates $\mathbf{q}$.

The parameterization of the rigid kinematics requires the definition of the relation between $\boldsymbol{\theta}$ and $\mathbf{q}$ (or $\mathbf{q}^*$) for the undeformed mechanism. The non-deformation condition is formulated using the *elastic potential energy* $\mathcal{V}(\mathbf{q})$ of the mechanism. An undeformed configuration satisfies the combined conditions:

$$\begin{cases} \frac{\partial \mathcal{V}}{\partial \mathbf{q}} = \mathbf{0} \\ \mathbf{\Phi}(\mathbf{q}) = \mathbf{0} \end{cases} \tag{5.38}$$

This problem can be analyzed numerically using the zero-strain approach. Accordingly, any *equilibrium configuration* achieves a minimum of the elastic potential energy, so that the kinematic problem becomes:

Given $\boldsymbol{\theta}$, find $\mathbf{q}^*$:

$$\min_{\mathbf{q}^*} \mathcal{V} \tag{5.39}$$

subject to

$$\mathbf{\Phi}(\boldsymbol{\theta}, \mathbf{q}^*) = \mathbf{0} \tag{5.40}$$

According to the augmented Lagrangian method, the following equivalent unconstrained problem is considered:

$$\min_{\mathbf{q}^*, \boldsymbol{\lambda}} \mathcal{V} + k\, \boldsymbol{\lambda}^T\, \mathbf{\Phi} + p\, \mathbf{\Phi}^T\, \mathbf{\Phi} \tag{5.41}$$

where $\boldsymbol{\lambda}$ is the $m \times 1$ vector of Lagrange multipliers, k and p are the scaling and penalty factors, respectively. The stationarity condition leads to $n + m$ nonlinear equations:

$$\begin{cases} \frac{\partial \mathcal{V}}{\partial \mathbf{q}^*} + \mathbf{\Phi}_{\mathbf{q}^*}^T\, (k\, \boldsymbol{\lambda} + p\, \mathbf{\Phi}) = \mathbf{0} \\ k\, \mathbf{\Phi}(\mathbf{q}) = \mathbf{0} \end{cases} \tag{5.42}$$

[2]The dimension here refers to the intrinsic dimension of the manifolds, as opposed to the dimension n of the ambient space.

Equations (5.42) contains as many equations as unknowns ($\mathbf{q}^*$,$\boldsymbol{\lambda}$), and can be solved using standard methods.

The initial problem was to establish $\mathbf{q}$ (or $\mathbf{q}^*$) from $\boldsymbol{\theta}$; it is equivalent to construct the relation between $\mathbf{u}$ (or $\mathbf{u}^*$) and $\boldsymbol{\theta}$, with:

$$\mathbf{u} = \begin{bmatrix} \boldsymbol{\theta} \\ \mathbf{q}^* \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\theta} \\ \mathbf{u}^* \end{bmatrix} \tag{5.43}$$

In the space of augmented coordinates $\mathbf{u}$, the total configuration space is denoted $\Omega^{rt}_{\mathbf{u}}$. Equation (5.42) gives a condition for static equilibrium which is a necessary but not sufficient condition for an undeformed configuration: some solutions $\mathbf{u}$ of (5.42) might not satisfy (5.38), and be out of $\Omega^{rt}_{\mathbf{u}}$. Typically, those solutions are associated with pre-stressed configurations ($\boldsymbol{\lambda} \neq \mathbf{0}$ or $\frac{\partial \mathcal{V}}{\partial \boldsymbol{\theta}} \neq \mathbf{0}$), which are not considered in this research.

If the Jacobian of the system (5.42) is not singular, the implicit function theorem guarantees the *local existence* of a function $\boldsymbol{\rho}$ such that $\mathbf{u} = \boldsymbol{\rho}(\boldsymbol{\theta})$. As in the rigid case, the *global validity* of the actuator parameterization can only be guaranteed in a restricted part of the configuration space $\Omega^{r}_{\mathbf{u}} \subset \Omega^{rt}_{\mathbf{u}}$, where the singularities are avoided. The parameterization mapping is defined by:

$$\boldsymbol{\rho} : \Omega^{r}_{\boldsymbol{\theta}} \rightarrow \Omega^{r}_{\mathbf{u}}, \qquad \boldsymbol{\theta} \mapsto \mathbf{u} = \boldsymbol{\rho}(\boldsymbol{\theta}) \tag{5.44}$$

where $\Omega^{r}_{\boldsymbol{\theta}} \subset \Omega^{rt}_{\boldsymbol{\theta}} \subset \mathcal{R}^s$ is the set of authorized variations of the actuated dofs $\boldsymbol{\theta}$.

Two important problems are associated with the practical computation of $\mathbf{u} = \boldsymbol{\rho}(\boldsymbol{\theta})$ and its Jacobian for a given $\boldsymbol{\theta}$.

Numerical computation of the kinematic mapping

For a given actuator configuration $\boldsymbol{\theta}$, the kinematic problem consists in determining the configuration $\mathbf{u} = \boldsymbol{\rho}(\boldsymbol{\theta})$. Starting from a known configuration $\mathbf{u}_{ref} = \boldsymbol{\rho}(\boldsymbol{\theta}_{ref})$, equations (5.42) can be solved for $\mathbf{q}^*$ and $\boldsymbol{\lambda}$ according to a standard Newton-Raphson procedure. As in section 3.2.2, the linearized equations are formulated:

$$\begin{bmatrix} \mathbf{K}_t^{\mathbf{q}^*\mathbf{q}^*} & k\,\boldsymbol{\Phi}^T_{\mathbf{q}^*} \\ k\,\boldsymbol{\Phi}_{\mathbf{q}^*} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{q}^* \\ \Delta\boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{5.45}$$

where $\mathbf{K}_t^{\mathbf{q}^*\mathbf{q}^*}$ is a submatrix of the tangent stiffness $\mathbf{K}_t$:

$$\mathbf{K}_t = \frac{\partial^2 \mathcal{V}}{\partial \mathbf{q}^2} + \frac{\partial \left(\mathbf{\Phi}_\mathbf{q}^T \left(k\, \boldsymbol{\lambda} + p\, \mathbf{\Phi} \right) \right)}{\partial \mathbf{q}} \tag{5.46}$$

$$= \frac{\partial^2 \mathcal{V}}{\partial \mathbf{q}^2} + p\, \mathbf{\Phi}_\mathbf{q}^T\, \mathbf{\Phi}_\mathbf{q} + k\, \frac{\partial \left(\mathbf{\Phi}_\mathbf{q}^T\, \boldsymbol{\lambda} \right)}{\partial \mathbf{q}} + p\, \sum_{i=1}^{m} \mathbf{\Phi}_i\, \frac{\partial^2 \mathbf{\Phi}_i}{\partial \mathbf{q}^2} \tag{5.47}$$

Usually, a good approximation is obtained by neglecting the second-order derivatives of the constraints:

$$\mathbf{K}_t \simeq \frac{\partial^2 \mathcal{V}}{\partial \mathbf{q}^2} + p\, \mathbf{\Phi}_\mathbf{q}^T\, \mathbf{\Phi}_\mathbf{q} \tag{5.48}$$

This approximation is exact at an undeformed configuration, since $\mathbf{\Phi} = \mathbf{0}$ and $\boldsymbol{\lambda} = \mathbf{0}$.

The converged solution might be out of $\Omega_\mathbf{u}^r$. This problem is avoided if the reference and the target configurations are sufficiently close to each other: the algorithm naturally converges to the unique solution in $\Omega_\mathbf{u}^r$, since it is also the unique solution in the neighborhood of the reference configuration. However, the danger is important nearby an actuator singularity, where different solution sets to the kinematic problem intersect with each other.

If the reference configuration $\boldsymbol{\theta}_{ref}$ is far from the target $\boldsymbol{\theta}$, the Newton-Raphson procedure is likely to diverge, or to converge to a solution outside $\Omega_\mathbf{u}^r$. Therefore, a linear homotopy path $\boldsymbol{\theta}_p : \mathcal{R} \to \mathcal{R}^s$ can be defined from $\boldsymbol{\theta}_{ref}$ to $\boldsymbol{\theta}$:

$$\boldsymbol{\theta}_p(\tau) = (1 - \tau)\, \boldsymbol{\theta}_{ref} + \tau\, \boldsymbol{\theta} \qquad \tau \in [0, 1] \tag{5.49}$$

The algorithm progresses from $\tau = 0$ to $\tau = 1$ through intermediate configurations. If a failure occurs, additional intermediate configurations are selected according to a bisection strategy, in order to obtain convergence. The procedure may not succeed if $\Omega_{\boldsymbol{\theta}}^r$ is non-convex; this rather unusual situation will not be detailed here, even though it could be handled by slight customization of the configuration space inspection algorithm described later in section 5.6.3.

Jacobian of the transformation

This section demonstrates the interpretation of the Jacobian $\boldsymbol{\rho}_{\boldsymbol{\theta}}$ as a set of rigid-body modes.

Any undeformed configuration satisfies the global static equilibrium:

$$\begin{cases} \frac{\partial \mathcal{V}}{\partial \mathbf{q}} + \mathbf{\Phi}_\mathbf{q}^T \left(k\, \boldsymbol{\lambda} + p\, \mathbf{\Phi} \right) = \mathbf{0} \\ k\, \mathbf{\Phi}(\mathbf{q}) = \mathbf{0} \end{cases} \tag{5.50}$$

This set of $n+m$ nonlinear equations can be recasted as

$$\mathbf{F}(\mathbf{u}) = \mathbf{0} \tag{5.51}$$

which is verified by any undeformed configuration:

$$\mathbf{F}(\boldsymbol{\rho}(\boldsymbol{\theta})) = \mathbf{0}, \qquad \forall\, \boldsymbol{\theta} \in \Omega^r_{\boldsymbol{\theta}} \tag{5.52}$$

After differentiation, we have

$$\mathbf{F}_{\mathbf{u}}\, \boldsymbol{\rho}_{\boldsymbol{\theta}} = \mathbf{0} \tag{5.53}$$

which can be developed:

$$\begin{bmatrix} \mathbf{K}_t & k\, \boldsymbol{\Phi}^T_{\mathbf{q}} \\ k\, \boldsymbol{\Phi}_{\mathbf{q}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{I} \\ \boldsymbol{\rho}^*_{\boldsymbol{\theta}} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{5.54}$$

This expression is equivalent to the definition of the rigid-body modes given by (5.18) for a linear structure. As a conclusion, *the Jacobian $\boldsymbol{\rho}_{\boldsymbol{\theta}}$ is the set of rigid-body modes of the linearized static equilibrium.*

5.4 The Global Modal Parameterization (GMP)

The concepts defined for the reduced-order parameterization of a deformable structure and for the minimal parameterization of the rigid kinematics are now combined for the reduced parameterization of a flexible mechanism. The Global Modal Parameterization results from the construction of a nonlinear coordinate transformation:

$$\overline{\mathbf{u}} = \boldsymbol{\chi}(\boldsymbol{\eta}) \tag{5.55}$$

where the $\overline{n}$ reduced coordinates $\boldsymbol{\eta}$ are defined in an open set $\Omega^f_{\boldsymbol{\eta}} \subset \mathcal{R}^{\overline{n}}$, and $\overline{\mathbf{u}} = \left[\mathbf{q}^T\ \boldsymbol{\lambda}^T\right]^T$ belongs to a submanifold of the flexible manifold: $\overline{\Omega}^f_{\mathbf{u}} \subset \Omega^f_{\mathbf{u}}$. The coordinate transformation is thus an invertible mapping:

$$\boldsymbol{\chi} : \Omega^f_{\boldsymbol{\eta}} \to \overline{\Omega}^f_{\mathbf{u}}, \qquad \boldsymbol{\eta} \mapsto \overline{\mathbf{u}} = \boldsymbol{\chi}(\boldsymbol{\eta}) \tag{5.56}$$

and the reduction comes from

$$\dim \overline{\Omega}^f_{\mathbf{u}} = \overline{n} < \dim \Omega^f_{\mathbf{u}} = n - m \tag{5.57}$$

The construction of the coordinate transformation $\boldsymbol{\chi}$ is based on a partitioned description of the rigid and of the flexible motion.

5.4.1 Partitioned coordinate transformation

Among the reduced coordinates $\boldsymbol{\eta}$, we dissociate a subset of s coordinates $\boldsymbol{\theta}$ for the description of the rigid kinematics, and a subset of $n^\delta = \overline{n} - s$ coordinates $\boldsymbol{\eta}^\delta$ for the description of the deformations:

$$\boldsymbol{\eta} = \begin{bmatrix} \boldsymbol{\theta} \\ \boldsymbol{\eta}^\delta \end{bmatrix} \tag{5.58}$$

$\boldsymbol{\theta}$ is the set of actuator dofs, able to parameterize the configuration space $\Omega^r_{\mathbf{u}}$. Under the assumption of small deformations, the reduction formula (5.55) is affine in $\boldsymbol{\eta}^\delta$:

$$\boxed{\overline{\mathbf{u}} = \boldsymbol{\rho}(\boldsymbol{\theta}) + \boldsymbol{\Psi}^{\mathbf{u}\delta}(\boldsymbol{\theta})\, \boldsymbol{\eta}^\delta} \tag{5.59}$$

$\boldsymbol{\rho}$ is the rigid kinematic mapping defined in section 5.3.2, and $\boldsymbol{\Psi}^{\mathbf{u}\delta}$ is a matrix of deformation modes which is still to be constructed, and which may depend on the configuration $\boldsymbol{\theta}$. The possible variations of the reduced coordinates are restricted to:

$$\boldsymbol{\theta} \in \Omega^r_{\boldsymbol{\theta}} \subset \mathcal{R}^s \qquad \text{and} \qquad \boldsymbol{\eta}^\delta \in \Omega^f_\delta(\boldsymbol{\theta}) \subset \mathcal{R}^{n^\delta} \tag{5.60}$$

where $\Omega^r_{\boldsymbol{\theta}}$ has been defined earlier and Ω^f_δ is a neighborhood of the origin.

Equation (5.59) can be differentiated:

$$\delta\overline{\mathbf{u}} = \boldsymbol{\chi}_{\boldsymbol{\eta}}\, \delta\boldsymbol{\eta} = \left(\boldsymbol{\rho}_{\boldsymbol{\theta}} + \frac{\partial(\boldsymbol{\Psi}^{\mathbf{u}\delta}\, \boldsymbol{\eta}^\delta)}{\partial \boldsymbol{\theta}} \right)\, \delta\boldsymbol{\theta} + \boldsymbol{\Psi}^{\mathbf{u}\delta}\, \delta\boldsymbol{\eta}^\delta \tag{5.61}$$

so that *for an undeformed configuration* ($\boldsymbol{\eta}^\delta = \mathbf{0}$):

$$\boldsymbol{\chi}_{\boldsymbol{\eta}} = \begin{bmatrix} \boldsymbol{\rho}_{\boldsymbol{\theta}} & \boldsymbol{\Psi}^{\mathbf{u}\delta} \end{bmatrix} = \boldsymbol{\Psi}^{\mathbf{u}\boldsymbol{\eta}} = \begin{bmatrix} \boldsymbol{\Psi}^{\mathbf{u}\boldsymbol{\theta}} & \boldsymbol{\Psi}^{\mathbf{u}\delta} \end{bmatrix} \tag{5.62}$$

This notation emphasizes the interpretation of the Jacobian $\boldsymbol{\rho}_{\boldsymbol{\theta}}$ as a set of rigid-body modes. For consistency, the Jacobian $\boldsymbol{\chi}_{\boldsymbol{\eta}}$ should belong to the tangent space of the flexible manifold $T_p\Omega^f_{\mathbf{u}}$; this property is important for the construction of the coordinate transformation, as discussed in the next section.

The reduced parameterization $\boldsymbol{\chi}$ of the flexible manifold involves the kinematic parameterization $\boldsymbol{\rho}$, which has been analyzed previously, and configuration-dependent flexible mode shapes $\boldsymbol{\Psi}^{\mathbf{u}\delta}$. In order to exploit component-mode techniques, the definition of this global parameterization starts with a local definition of mode shapes around an undeformed configuration. The connection of this local definition with the global parameterization will be demonstrated afterwards.

5.4.2 Definition of the local modes

For a given configuration, the construction of the mode shapes is based on the linearized dynamic equations. As for the linear component-mode technique, a partitioning of the initial dofs $\mathbf{u}$ is considered

$$\mathbf{u} = \begin{bmatrix} \boldsymbol{\theta} \\ \mathbf{q}^g \\ \mathbf{u}^i \end{bmatrix} \tag{5.63}$$

where $\boldsymbol{\theta}$ are the actuator dofs, $\mathbf{q}^g$ are the constraint dofs, and $\mathbf{u}^i$ are the internal dofs, including the Lagrange multipliers.

The mode shapes are selected as an optimal basis to describe the linearized motion around an *undeformed configuration*, with *zero velocities*:

$$\begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\ddot{\mathbf{q}} \\ \Delta\ddot{\boldsymbol{\lambda}} \end{bmatrix} + \begin{bmatrix} \mathbf{K}_t & k\,\boldsymbol{\Phi}_{\mathbf{q}}^T \\ k\,\boldsymbol{\Phi}_{\mathbf{q}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{q} \\ \Delta\boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{5.64}$$

The approximated expression of the tangent stiffness (5.48) is thus exact and can be safely exploited.

As in the linear case, three subsets of modes are defined:

$$\boldsymbol{\Psi}^{\mathbf{u}\boldsymbol{\eta}} = \begin{bmatrix} \boldsymbol{\Psi}^{\mathbf{u}\boldsymbol{\theta}} & \boldsymbol{\Psi}^{\mathbf{u}\gamma} & \boldsymbol{\Psi}^{\mathbf{u}\iota} \end{bmatrix} \tag{5.65}$$

The rigid modes $\boldsymbol{\Psi}^{\mathbf{u}\boldsymbol{\theta}}$ are defined from the static equilibrium (5.18); they are also the Jacobian of the rigid kinematics: $\boldsymbol{\Psi}^{\mathbf{u}\boldsymbol{\theta}} = \boldsymbol{\rho}_{\boldsymbol{\theta}}$. The constraint modes $\boldsymbol{\Psi}^{\mathbf{u}\gamma}$ are the solutions to the static problem (5.19), where the rigid dofs are fixed, and the internal modes $\boldsymbol{\Psi}^{\mathbf{u}\iota}$ are defined according to the internal eigenvalue problem (5.20), where the rigid and constraint dofs are fixed.

For local consistency, those modes should belong to the tangent space of the flexible manifold at point p: $T_p\Omega_{\mathbf{u}}^f$. Indeed, using (5.18), (5.19), (5.20) and the structure of the tangent stiffness matrix, it is easy to verify that

$$\boldsymbol{\Phi}_{\mathbf{u}}\ \boldsymbol{\Psi}^{\mathbf{u}\boldsymbol{\eta}} = \mathbf{0} \tag{5.66}$$

where $\boldsymbol{\Phi}_{\mathbf{u}} = [\boldsymbol{\Phi}_{\mathbf{q}} \quad \mathbf{0}]$.

Figures 5.5, 5.6 and 5.7 illustrate the definition of the local modes for a few simple situations.

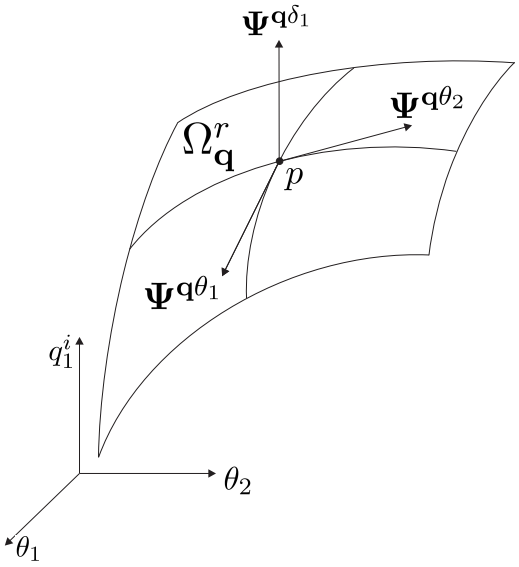


Figure 5.5: Local modes of an unconstrained mechanism : 2 rigid modes and one flexible mode.

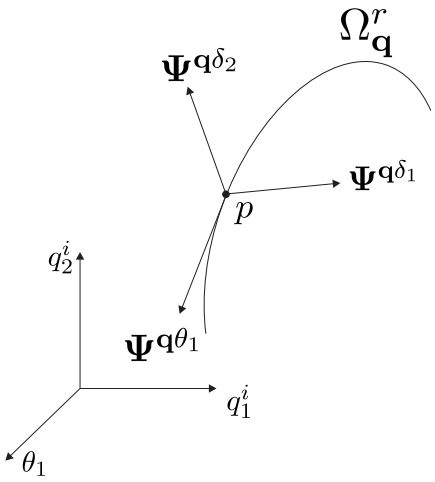


Figure 5.6: Local modes of an unconstrained mechanism: one rigid mode and two flexible modes.

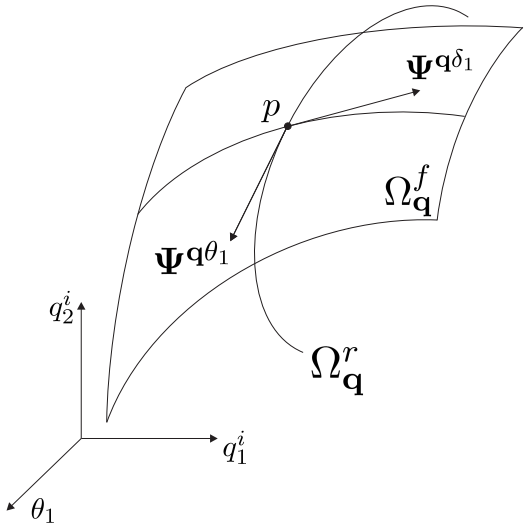


Figure 5.7: Local modes of a constrained mechanism: one rigid mode, and one flexible mode.

5.4.3 From local to global modes

The local definition of the flexible modes can be reproduced for any configuration, leading to configuration-dependent mode shapes. *A priori*, nothing prevents spurious discontinuous variations of those mode shapes, and it is necessary to verify that the Global Modal Parameterization $\boldsymbol{\chi}(\boldsymbol{\eta})$ is well-defined, *i.e.*

- it is globally one-to-one,
- it is continuously differentiable,
- its rank is maximal (*i.e.* the Jacobian $\boldsymbol{\chi}_{\boldsymbol{\eta}}$ is of rank $\overline{n}$),

From the previous developments, $\boldsymbol{\chi}(\boldsymbol{\eta})$ involves two contributions:

- the kinematic mapping $\boldsymbol{\rho}(\boldsymbol{\theta})$, which is well-defined in the configuration space of interest, as demonstrated in section 5.3; in particular, it is continuous and differentiable,
- the flexible modes $\boldsymbol{\Psi}^{\mathbf{u}^\delta}(\boldsymbol{\theta})$ have only received a local definition at a given configuration; their variations in the configuration space deserve a careful analysis.

The following analysis demonstrates that the three consistency criteria are satisfied by the Global Modal Parameterization.

One-to-one criterion

The flexible submanifold $\overline{\Omega}^f_{\mathbf{u}}$ is defined as the image of $\Omega^f_{\boldsymbol{\eta}}$ mapped by the transformation $\boldsymbol{\chi}$. By construction, for all $\overline{\mathbf{u}} \in \overline{\Omega}^f_{\mathbf{u}}$, there exists a vector $\boldsymbol{\eta}$ such that $\overline{\mathbf{u}} = \boldsymbol{\chi}(\boldsymbol{\eta})$. Our purpose is to demonstrate the uniqueness of such a vector $\boldsymbol{\eta}$ in $\Omega^f_{\boldsymbol{\eta}}$.

Suppose that another solution $\boldsymbol{\eta}'$ also satisfies $\overline{\mathbf{u}} = \boldsymbol{\chi}(\boldsymbol{\eta}')$. Since the actuator dofs $\boldsymbol{\theta}$ appear among the mixed coordinates, we necessarily have $\boldsymbol{\theta}' = \boldsymbol{\theta}$, and the flexible coordinates satisfy

$$\boldsymbol{\rho}(\boldsymbol{\theta}) + \boldsymbol{\Psi}^{\mathbf{u}\delta}(\boldsymbol{\theta})\, \boldsymbol{\eta}^{\delta} = \boldsymbol{\rho}(\boldsymbol{\theta}) + \boldsymbol{\Psi}^{\mathbf{u}\delta}(\boldsymbol{\theta})\, \boldsymbol{\eta}^{\delta'} \quad \Rightarrow \quad \boldsymbol{\Psi}^{\mathbf{u}\delta}\, (\boldsymbol{\eta}^{\delta} - \boldsymbol{\eta}^{\delta'}) = \mathbf{0} \tag{5.67}$$

From the linear independence of the flexible modes, we deduce that $\boldsymbol{\eta}^{\delta} - \boldsymbol{\eta}^{\delta'} = \mathbf{0}$, which completes the proof.

Continuity of the constraint modes

The constraint modes $\boldsymbol{\Psi}^{\mathbf{u}\gamma}$ are defined from the linear static analysis when the rigid dofs are fixed, see equation (5.19), and we have

$$\boldsymbol{\Psi}^{i\gamma}(\boldsymbol{\theta}) = -\left(\mathbf{K}^{ii}(\boldsymbol{\theta})\right)^{-1}\, \mathbf{K}^{ig}(\boldsymbol{\theta}) \tag{5.68}$$

Since $\mathbf{K}^{ii}$ is invertible[3], and since all the stiffness coefficients are continuous and differentiable, so are the constraint modes.

Continuity of the internal modes

The internal modes $\boldsymbol{\Psi}^{i\iota}$ are obtained after three steps:

1. the eigenvalue problem is solved:

$$\left(\mathbf{K}^{ii}(\boldsymbol{\theta}) - \omega^2\, \mathbf{M}^{ii}(\boldsymbol{\theta})\right)\, \boldsymbol{\Psi}^{i\iota *}(\boldsymbol{\theta}) = \mathbf{0} \tag{5.69}$$

2. the resulting $n^{\iota *}$ eigenmodes are sorted according to their eigenvalue, and the first $n^{\iota} < n^{\iota *}$ modes are selected,

3. the modes are normalized with respect to the mass matrix so that:

$$\boldsymbol{\Psi}^{i\iota T}\, \mathbf{M}^{ii}\, \boldsymbol{\Psi}^{i\iota} = \mathbf{I}^{\iota\iota} \tag{5.70}$$

[3]Otherwise, additional rigid modes should be defined.

For one selected eigenvalue ω_k and the associated internal mode $\boldsymbol{\psi}_k$, it is instructive to differentiate (5.69) with respect to a coordinate θ_l:

$$\left(\frac{\partial \mathbf{K}^{ii}}{\partial \theta_l} - \omega_k^2 \, \frac{\partial \mathbf{M}^{ii}}{\partial \theta_l} - \frac{\partial \omega_k^2}{\partial \theta_l} \, \mathbf{M}^{ii} \right) \boldsymbol{\psi}_k + \left(\mathbf{K}^{ii} - \omega_k^2 \, \mathbf{M}^{ii} \right) \frac{\partial \boldsymbol{\psi}_k}{\partial \theta_l} = \mathbf{0} \tag{5.71}$$

A pre-multiplication by $\boldsymbol{\psi}_k^T$ leads to the extraction of $\frac{\partial \omega_k^2}{\partial \theta_l}$:

$$\frac{\partial \omega_k^2}{\partial \theta_l} = \frac{1}{\mu_k} \, \boldsymbol{\psi}_k^T \left(\frac{\partial \mathbf{K}^{ii}}{\partial \theta_l} - \omega_k^2 \, \frac{\partial \mathbf{M}^{ii}}{\partial \theta_l} \right) \boldsymbol{\psi}_k \tag{5.72}$$

with the modal mass $\mu_k = \boldsymbol{\psi}_k^T \, \mathbf{M}^{ii} \, \boldsymbol{\psi}_k > 0$. Therefore, each eigenvalue ω_k exhibits smooth variations with respect to the coordinate θ_l.

From the continuity of ω_k, and of the mass and stiffness matrices, the first term in (5.71) is continuous, and so is the second one. However, the derivatives of the eigenmodes $\frac{\partial \boldsymbol{\psi}_k}{\partial \theta_l}$ can take arbitrary large values in the kernel of $\mathbf{K}^{ii} - \omega_k^2 \, \mathbf{M}^{ii}$.

If ω_k has a multiplicity $\kappa = 1$, $\boldsymbol{\psi}_k$ is the single vector in this kernel, which means that the arbitrary variations are parallel to the initial vector. Those variations are forbidden due to the normalization condition (5.70), and we conclude that $\frac{\partial \boldsymbol{\psi}_k}{\partial \theta_l}$ is continuous.

This property is no more guaranteed for a multiplicity $\kappa > 1$. In this case, the kernel of $\mathbf{K}^{ii} - \omega_k^2 \, \mathbf{M}^{ii}$ can be a multi-dimensional subspace, where the eigenvectors can take arbitrary variations.

An example is given in Figure 5.8. In the one-dimensional configuration space, there is one configuration $\theta_1 = \theta^b$ for which $\omega_1 = \omega_2$ and $\kappa = 2$. For $\theta_1 < \theta^b$ or for $\theta_1 > \theta^b$, $\boldsymbol{\psi}_1$ and $\boldsymbol{\psi}_2$ are thus continuous. By inspection of equation (5.69), and considering that the eigenvalues are continuous across θ^b, an eigenmode found on one side of the boundary θ^b is still a valid eigenmode on the other side. For this reason, it is possible to impose $\Delta\boldsymbol{\psi}_1 = \mathbf{0}$ and $\Delta\boldsymbol{\psi}_2 = \mathbf{0}$ from one side of θ^b to the other, leading to a continuous parameterization.

This analysis is naturally extended for a s-dimensional configuration space. The domains such that $\kappa > 1$ are b-dimensional manifold $\mathcal{M}^b$, with $b < s$ (the less usual case where $b = s$ deserves a specific treatment, which is not considered here). It is still possible to impose $\Delta\boldsymbol{\psi}_k = \mathbf{0}$ across the boundary $\mathcal{M}^b$, which motivates the implementation of a *mode tracking strategy* at the step 2 of the construction of the internal modes.

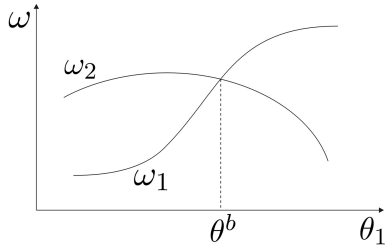


Figure 5.8: Variations of the eigenfrequencies in the configuration space.

Similar mode-tracking problems arise in topological optimization problems; for instance, Kim and Kim [KK00] have proposed a method based on the Modal Assurance Criterion (MAC value) defined in structural identification. The MAC number is a measure from 0 to 1 of the correlation between two modes, a unitary value meaning perfect correlation. Our method relies on a correlation of the $n^{\iota *}$ eigenmodes computed at the current configuration, with n^{ι} eigenmodes $\boldsymbol{\Psi}^{i\iota}_{ref}$ associated with a reference configuration. A mode matching algorithm defines the eigenmodes $\boldsymbol{\Psi}^{i\iota}$ in order to obtain MAC numbers as close as possible to 1. More precisely, the MAC matrix, which is filled by the MAC numbers of all pairs of modes, is rendered as close as possible to the identity.

Ideally, all diagonal terms of the resulting MAC matrix are above 0.95, and the consistency of the internal modes is guaranteed. If the reference configuration is far from the current configuration, lower MAC values may be obtained, so that the internal modes $\boldsymbol{\Psi}^{i\iota}$ cannot be validated. Then, the algorithm should restart the procedure with a closer reference configuration. Therefore, the mode-tracking problem is partly handled by the configuration space inspection algorithm, which will be explained in section 5.6.3.

Rank of the Jacobian

The last criterion for a well-defined parameterization is that the Jacobian $\boldsymbol{\chi}_{\boldsymbol{\eta}}$ has full rank. At an undeformed configuration, the columns of $\boldsymbol{\chi}_{\boldsymbol{\eta}}$ are the local modes, see equation (5.62). By construction, their linear independence is guaranteed, and the rank condition is satisfied. At a deformed configuration, the expression of the Jacobian is slightly affected (5.61), but under the assumption of small deformations, the full-rank requirement is still verified.

From this analysis, we conclude that *the Global Modal Parameterization is well-defined, provided that a mode-tracking strategy is implemented.* Hence, it is a reliable basis for the formulation of a reduced-order model.

5.5 Reduced-order model

From the definition of the Global Modal Parameterization, we shall now formulate the equations of motion in terms of the modal parameters. The Lagrange equations involve contributions from the elastic potential energy (elastic forces), the kinetic energy (inertia forces), and the work produced by external forces.

5.5.1 Elastic forces

The elastic potential energy is a strongly nonlinear function $\mathcal{V}(\mathbf{q})$, and we seek for an approximation in terms of the reduced coordinates $\boldsymbol{\eta}$. From the dissociation between rigid and flexible coordinates, a smoother behavior is expected for $\mathcal{V}(\boldsymbol{\theta}, \boldsymbol{\eta}^\delta)$. Around an undeformed configuration p characterized by $\boldsymbol{\theta} = \boldsymbol{\theta}_0$ and $\boldsymbol{\eta}^\delta = \mathbf{0}$, let us consider the order 2 Taylor series expansion:

$$\mathcal{V}(\boldsymbol{\eta}) = \mathcal{V}_p + \left(\frac{\partial \mathcal{V}}{\partial \boldsymbol{\eta}}\right)_p^T \Delta\boldsymbol{\eta} + \frac{1}{2}\,\Delta\boldsymbol{\eta}^T \left(\frac{\partial^2 \mathcal{V}}{\partial \boldsymbol{\eta}^2}\right)_p \Delta\boldsymbol{\eta} \tag{5.73}$$

with

$$\Delta\boldsymbol{\eta} = \begin{bmatrix} \boldsymbol{\theta} - \boldsymbol{\theta}_0 \\ \boldsymbol{\eta}^\delta \end{bmatrix} \tag{5.74}$$

At the undeformed configuration, the elastic potential energy is zero $\mathcal{V}_p = 0$, and the generalized elastic forces vanish $\left(\frac{\partial \mathcal{V}}{\partial \boldsymbol{\eta}}\right)_p = \mathbf{0}$.

The second order derivatives of $\mathcal{V}$ is an equivalent stiffness

$$\overline{\mathbf{K}}^{\eta\eta} = \frac{\partial^2 \mathcal{V}}{\partial \boldsymbol{\eta}^2} \tag{5.75}$$

so that

$$\mathcal{V} = \frac{1}{2}\,\Delta\boldsymbol{\eta}^T \left(\overline{\mathbf{K}}^{\eta\eta}\right)_p \Delta\boldsymbol{\eta} \tag{5.76}$$

The objective is to establish a connection between the reduced stiffness $(\overline{\mathbf{K}}^{\eta\eta})_p$ and the initial Finite Element formulation. All the following developments are re-

alized *at the undeformed configuration p*. Let us analyze the second-order derivative of the augmented functional $\mathcal{V}^* = \mathcal{V} + k\,\boldsymbol{\lambda}^T\,\boldsymbol{\Phi} + p\,\boldsymbol{\Phi}^T\,\boldsymbol{\Phi}$:

$$\frac{\partial^2 \mathcal{V}^*}{\partial \boldsymbol{\eta}^2} = \boldsymbol{\Psi}^{\mathbf{u}\boldsymbol{\eta}T}\,\frac{\partial^2 \mathcal{V}^*}{\partial \mathbf{u}^2}\,\boldsymbol{\Psi}^{\mathbf{u}\boldsymbol{\eta}} \tag{5.77}$$

$$= \begin{bmatrix} \boldsymbol{\Psi}^{\mathbf{q}\boldsymbol{\eta}T} & \boldsymbol{\Psi}^{\boldsymbol{\lambda}\boldsymbol{\eta}T} \end{bmatrix} \begin{bmatrix} \mathbf{K}_t & k\,\boldsymbol{\Phi}_{\mathbf{q}}^T \\ k\,\boldsymbol{\Phi}_{\mathbf{q}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Psi}^{\mathbf{q}\boldsymbol{\eta}} \\ \boldsymbol{\Psi}^{\boldsymbol{\lambda}\boldsymbol{\eta}} \end{bmatrix} \tag{5.78}$$

$$= \boldsymbol{\Psi}^{\mathbf{q}\boldsymbol{\eta}T}\,\mathbf{K}_t\,\boldsymbol{\Psi}^{\mathbf{q}\boldsymbol{\eta}} \tag{5.79}$$

The property $\boldsymbol{\Phi}_{\mathbf{q}}\,\boldsymbol{\Psi}^{\mathbf{q}\boldsymbol{\eta}} = \mathbf{0}$ has been used. Since $\mathbf{K}_t = \frac{\partial^2 \mathcal{V}}{\partial \mathbf{q}^2} + p\,\boldsymbol{\Phi}_{\mathbf{q}}^T\,\boldsymbol{\Phi}_{\mathbf{q}}$, we have:

$$\boldsymbol{\Psi}^{\mathbf{q}\boldsymbol{\eta}T}\,\mathbf{K}_t\,\boldsymbol{\Psi}^{\mathbf{q}\boldsymbol{\eta}} = \boldsymbol{\Psi}^{\mathbf{q}\boldsymbol{\eta}T}\,\frac{\partial^2 \mathcal{V}}{\partial \mathbf{q}^2}\,\boldsymbol{\Psi}^{\mathbf{q}\boldsymbol{\eta}} = \frac{\partial^2 \mathcal{V}}{\partial \boldsymbol{\eta}^2} \tag{5.80}$$

and we conclude:

$$\overline{\mathbf{K}}^{\boldsymbol{\eta}\boldsymbol{\eta}} = \boldsymbol{\Psi}^{\mathbf{q}\boldsymbol{\eta}T}\,\mathbf{K}_t\,\boldsymbol{\Psi}^{\mathbf{q}\boldsymbol{\eta}} \tag{5.81}$$

Since by construction of the rigid modes $\mathbf{K}_t\,\boldsymbol{\Psi}^{\mathbf{q}\boldsymbol{\theta}} = \mathbf{0}$, the equivalent stiffness has the structure

$$\overline{\mathbf{K}}^{\boldsymbol{\eta}\boldsymbol{\eta}} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \overline{\mathbf{K}}^{\delta\delta} \end{bmatrix} \tag{5.82}$$

with

$$\overline{\mathbf{K}}^{\delta\delta} = \boldsymbol{\Psi}^{\mathbf{q}\delta T}\,\mathbf{K}_t\,\boldsymbol{\Psi}^{\mathbf{q}\delta} \tag{5.83}$$

This last formula can be directly exploited in the Finite Element code to compute the equivalent stiffness. Finally, the potential energy is represented by a quadratic form in the flexible coordinates:

$$\mathcal{V}(\boldsymbol{\theta}, \boldsymbol{\eta}^\delta) = \frac{1}{2}\,\boldsymbol{\eta}^{\delta T}\,\overline{\mathbf{K}}^{\delta\delta}(\boldsymbol{\theta})\,\boldsymbol{\eta}^\delta \tag{5.84}$$

The generalized elastic forces are:

$$\frac{\partial \mathcal{V}}{\partial \boldsymbol{\eta}^\delta} = \overline{\mathbf{K}}^{\delta\delta}(\boldsymbol{\theta})\,\boldsymbol{\eta}^\delta \tag{5.85}$$

The contribution $\frac{\partial \mathcal{V}}{\partial \boldsymbol{\theta}}$ is quadratic in the amplitude of deformation; it is neglected under the small deformation assumption.

5.5.2 Inertia forces

In the initial model, the kinetic energy is a quadratic form of the velocities

$$\mathcal{K} = \frac{1}{2}\,\dot{\mathbf{q}}^T\,\mathbf{M}\,\dot{\mathbf{q}} \tag{5.86}$$

Considering the coordinate transformation at the velocity level

$$\dot{\overline{\mathbf{q}}} = \boldsymbol{\chi}_{\eta}\, \dot{\boldsymbol{\eta}} \tag{5.87}$$

we obtain:

$$\mathcal{K} = \frac{1}{2}\, \dot{\boldsymbol{\eta}}^T\, \overline{\mathbf{M}}^{\eta\eta}\, \dot{\boldsymbol{\eta}} \qquad \text{with} \qquad \overline{\mathbf{M}}^{\eta\eta} = \boldsymbol{\chi}_{\eta}^T\, \mathbf{M}\, \boldsymbol{\chi}_{\eta} \tag{5.88}$$

The generalized inertia forces are defined by:

$$\overline{\mathbf{g}}_{iner} = \frac{d}{dt}\left(\frac{\partial \mathcal{K}}{\partial \dot{\boldsymbol{\eta}}}\right) - \frac{\partial \mathcal{K}}{\partial \boldsymbol{\eta}} \tag{5.89}$$

A *simplified expression* is possible if the variations of the reduced mass matrix $\overline{\mathbf{M}}^{\eta\eta}$ are neglected:

$$\overline{\mathbf{g}}_{iner} = \overline{\mathbf{M}}^{\eta\eta}\, \ddot{\boldsymbol{\eta}} \tag{5.90}$$

For a more thorough analysis, let us consider a partitioning of the generalized coordinates into translation and rotation dofs:

$$\mathbf{q} = \left[\begin{array}{c} \mathbf{x} \\ \boldsymbol{\psi} \end{array}\right] = \left[\begin{array}{c} \boldsymbol{\chi}^{\mathbf{x}}(\boldsymbol{\eta}) \\ \boldsymbol{\chi}^{\psi}(\boldsymbol{\eta}) \end{array}\right] \tag{5.91}$$

In the formulation proposed by Géradin and Cardona [GC01], their contributions to the kinetic energy are uncoupled:

$$\mathcal{K} = \mathcal{K}^{\mathbf{x}} + \mathcal{K}^{\psi} = \frac{1}{2}\, \dot{\mathbf{x}}^T\, \mathbf{M}^{\mathbf{xx}}\, \dot{\mathbf{x}} + \frac{1}{2}\, \dot{\boldsymbol{\psi}}^T\, \mathbf{M}^{\psi\psi}\, \dot{\boldsymbol{\psi}} \tag{5.92}$$

$\mathbf{M}^{\mathbf{xx}}$ is constant, but $\mathbf{M}^{\psi\psi}$ is subject to variations due to the updated Lagrangian point of view adopted for the rotation parameters.

Inertia forces associated with the translation kinetic energy

The contribution of the translation kinetic energy to the generalized inertia forces is:

$$\overline{\mathbf{g}}_{iner}^{trans} = \frac{d}{dt}\left(\frac{\partial \mathcal{K}^{\mathbf{x}}}{\partial \dot{\boldsymbol{\eta}}}\right) - \frac{\partial \mathcal{K}^{\mathbf{x}}}{\partial \boldsymbol{\eta}} \tag{5.93}$$

$$= \left(\frac{\partial \dot{\mathbf{x}}}{\partial \dot{\boldsymbol{\eta}}}\right)^T \mathbf{M}^{\mathbf{xx}}\, \ddot{\mathbf{x}} + \left[\frac{d}{dt}\left(\frac{\partial \dot{\mathbf{x}}}{\partial \dot{\boldsymbol{\eta}}}\right) - \left(\frac{\partial \dot{\mathbf{x}}}{\partial \boldsymbol{\eta}}\right)\right]^T \mathbf{M}^{\mathbf{xx}}\, \dot{\mathbf{x}} \tag{5.94}$$

We also have

$$\dot{\mathbf{x}} = \boldsymbol{\chi}_{\eta}^{\mathbf{x}}\, \dot{\boldsymbol{\eta}} \qquad \text{and} \qquad \ddot{\mathbf{x}} = \dot{\boldsymbol{\chi}}_{\eta}^{\mathbf{x}}\, \dot{\boldsymbol{\eta}} + \boldsymbol{\chi}_{\eta}^{\mathbf{x}}\, \ddot{\boldsymbol{\eta}} \tag{5.95}$$

so that

$$\overline{\mathbf{g}}_{iner}^{trans} = \boldsymbol{\chi}_{\boldsymbol{\eta}}^{\mathbf{x}T}\,\mathbf{M}^{\mathbf{xx}}\,\boldsymbol{\chi}_{\boldsymbol{\eta}}^{\mathbf{x}}\,\ddot{\boldsymbol{\eta}} + \boldsymbol{\chi}_{\boldsymbol{\eta}}^{\mathbf{x}T}\,\mathbf{M}^{\mathbf{xx}}\,\dot{\boldsymbol{\chi}}_{\boldsymbol{\eta}}^{\mathbf{x}}\,\dot{\boldsymbol{\eta}} + \left[\dot{\boldsymbol{\chi}}_{\boldsymbol{\eta}}^{\mathbf{x}} - \frac{\partial(\boldsymbol{\chi}_{\boldsymbol{\eta}}^{\mathbf{x}}\,\dot{\boldsymbol{\eta}})}{\partial\boldsymbol{\eta}}\right]^T \mathbf{M}^{\mathbf{xx}}\,\boldsymbol{\chi}_{\boldsymbol{\eta}}^{\mathbf{x}}\,\dot{\boldsymbol{\eta}} \quad (5.96)$$

Using the index summation convention, we develop:

$$\left(\dot{\boldsymbol{\chi}}_{\boldsymbol{\eta}}^{\mathbf{x}}\right)_{ij} = \frac{d}{dt}\left(\frac{\partial\chi_i^{\mathbf{x}}}{\partial\eta_j}\right) = \frac{\partial^2\chi_i^{\mathbf{x}}}{\partial\eta_k\partial\eta_j}\,\dot{\eta}_k \quad (5.97)$$

$$\left(\frac{\partial(\boldsymbol{\chi}_{\boldsymbol{\eta}}^{\mathbf{x}}\,\dot{\boldsymbol{\eta}})}{\partial\boldsymbol{\eta}}\right)_{ij} = \frac{\partial\left(\dot{\boldsymbol{\chi}}_{\boldsymbol{\eta}}^{\mathbf{x}}\,\dot{\boldsymbol{\eta}}\right)_i}{\partial\eta_j} = \frac{\partial}{\partial\eta_j}\left(\frac{\partial\chi_i^{\mathbf{x}}}{\partial\eta_k}\,\dot{\eta}_k\right) = \frac{\partial^2\chi_i^{\mathbf{x}}}{\partial\eta_j\partial\eta_k}\,\dot{\eta}_k \quad (5.98)$$

Thus, the last term in (5.96) vanishes. It is convenient to define the third order *curvature tensor* $\boldsymbol{\Gamma}^{\boldsymbol{\chi}^{\mathbf{x}}}$:

$$\Gamma_{ijk}^{\boldsymbol{\chi}^{\mathbf{x}}} = \frac{\partial^2\chi_i^{\mathbf{x}}}{\partial\eta_j\,\partial\eta_k} \qquad \text{with} \qquad \Gamma_{ijk}^{\boldsymbol{\chi}^{\mathbf{x}}} = \Gamma_{ikj}^{\boldsymbol{\chi}^{\mathbf{x}}} \quad (5.99)$$

so that

$$\left(\dot{\boldsymbol{\chi}}_{\boldsymbol{\eta}}^{\mathbf{x}}\right)_{ij} = \Gamma_{ijk}^{\boldsymbol{\chi}^{\mathbf{x}}}\,\dot{\eta}_k = \left(\boldsymbol{\Gamma}^{\boldsymbol{\chi}^{\mathbf{x}}}.\,\dot{\boldsymbol{\eta}}\right)_{ij} \quad (5.100)$$

where the "." operator is defined as a product of a third order tensor by a vector. Hence, the translation inertia forces are concisely formulated

$$\overline{\mathbf{g}}_{iner}^{trans} = \boldsymbol{\chi}_{\boldsymbol{\eta}}^{\mathbf{x}T}\,\mathbf{M}^{\mathbf{xx}}\,\boldsymbol{\chi}_{\boldsymbol{\eta}}^{\mathbf{x}}\,\ddot{\boldsymbol{\eta}} + \boldsymbol{\chi}_{\boldsymbol{\eta}}^{\mathbf{x}T}\,\mathbf{M}^{\mathbf{xx}}\,\left(\boldsymbol{\Gamma}^{\boldsymbol{\chi}^{\mathbf{x}}}.\,\dot{\boldsymbol{\eta}}\right)\,\dot{\boldsymbol{\eta}} \quad (5.101)$$

This expression is composed of two terms:

- the inertia forces obtained when the variations of the reduced mass matrix are quasi-static,
- the equivalent forces associated with the curvature (or the nonlinearity) of the coordinate reduction formula.

Inertia forces associated with the rotation kinetic energy of a rigid body

The rotation kinetic energy of an isolated rigid body A is expressed by:

$$\mathcal{K}^{\psi,A} = \frac{1}{2}\,\boldsymbol{\Omega}^T\,\mathbf{J}\,\boldsymbol{\Omega} \quad (5.102)$$

where $\boldsymbol{\Omega}$ is the 3×1 vector of material angular velocities, and $\mathbf{J}$ is the inertia tensor. According to Géradin and Cardona [GC01], $\boldsymbol{\Omega}$ is connected to the orientation parameters $\boldsymbol{\alpha}$ of the body by the 3×3 tangent operator $\mathbf{T}$ of the rotation parameterization:

$$\boldsymbol{\Omega} = \mathbf{T}(\boldsymbol{\alpha})\,\dot{\boldsymbol{\alpha}} \quad (5.103)$$

In the Finite Element formulation, an updated Lagrangian point of view is adopted, and the reduction procedure is considered at an updated configuration $\boldsymbol{\alpha} = \mathbf{0}$. If the rotations are parameterized using the Cartesian rotation vector or the conformal rotation vector, we have the useful properties:

$$T_{ij}(\mathbf{0}) = \delta_{ij} \qquad \text{and} \qquad \frac{\partial T_{ij}}{\partial \alpha_k}(\mathbf{0}) = \frac{1}{2}\,\epsilon_{ijk} \tag{5.104}$$

where δ_{ij} is the Kronecker symbol, and ϵ_{ijk} is the alternating symbol, defined by

$$\epsilon_{ijk} = \begin{cases} 1 \text{ if } (ijk) \text{ is a cyclic permutation of } (123), \\ -1 \text{ if } (ijk) \text{ is an anticyclic permutation of } (123), \\ 0 \text{ otherwise.} \end{cases} \tag{5.105}$$

The parameters $\boldsymbol{\alpha}$ are a subset of the orientation parameters $\boldsymbol{\psi}$. If we define $\boldsymbol{\chi}^{\boldsymbol{\alpha}} = \mathbf{L}^{\boldsymbol{\alpha\psi}}\,\boldsymbol{\chi}^{\boldsymbol{\psi}}$, where $\mathbf{L}^{\boldsymbol{\alpha\psi}}$ is a localization matrix, we have:

$$\boldsymbol{\alpha} = \boldsymbol{\chi}^{\boldsymbol{\alpha}}(\boldsymbol{\eta}) \qquad \text{and} \qquad \dot{\boldsymbol{\alpha}} = \boldsymbol{\chi}^{\boldsymbol{\alpha}}_{\boldsymbol{\eta}}\,\dot{\boldsymbol{\eta}} \tag{5.106}$$

Introducing the operator $\boldsymbol{\chi}^{\boldsymbol{\Omega\eta}}$:

$$\boldsymbol{\chi}^{\boldsymbol{\Omega\eta}} = \mathbf{T}\,\boldsymbol{\chi}^{\boldsymbol{\alpha}}_{\boldsymbol{\eta}} \tag{5.107}$$

we have

$$\boldsymbol{\Omega} = \boldsymbol{\chi}^{\boldsymbol{\Omega\eta}}\,\dot{\boldsymbol{\eta}} \tag{5.108}$$

and the developments realized for the translation kinetic energy can be reproduced:

$$\overline{\mathbf{g}}^{rot,A}_{iner} = \boldsymbol{\chi}^{\boldsymbol{\Omega\eta}^T}\,\mathbf{J}\,\boldsymbol{\chi}^{\boldsymbol{\Omega\eta}}\,\ddot{\boldsymbol{\eta}} + \boldsymbol{\chi}^{\boldsymbol{\Omega\eta}^T}\,\mathbf{J}\,\dot{\boldsymbol{\chi}}^{\boldsymbol{\Omega\eta}}\,\dot{\boldsymbol{\eta}} + \left[\dot{\boldsymbol{\chi}}^{\boldsymbol{\Omega\eta}} - \frac{\partial(\boldsymbol{\chi}^{\boldsymbol{\Omega\eta}}\,\dot{\boldsymbol{\eta}})}{\partial \boldsymbol{\eta}}\right]^T \mathbf{J}\,\boldsymbol{\chi}^{\boldsymbol{\Omega\eta}}\,\dot{\boldsymbol{\eta}} \tag{5.109}$$

$\boldsymbol{\chi}^{\boldsymbol{\Omega\eta}}$ is not a Jacobian, and the last term does not vanish in this case. Let us develop:

$$\left(\dot{\boldsymbol{\chi}}^{\boldsymbol{\Omega\eta}}\right)_{ij} = \left(\frac{\partial T_{iq}}{\partial \eta_k}\,\frac{\partial \chi^{\boldsymbol{\alpha}}_q}{\partial \eta_j} + T_{iq}\,\frac{\partial^2 \chi^{\boldsymbol{\alpha}}_q}{\partial \eta_j\,\partial \eta_k}\right)\,\dot{\eta}_k \tag{5.110}$$

$$\frac{\partial}{\partial \boldsymbol{\eta}_j}\left(\boldsymbol{\chi}^{\boldsymbol{\Omega\eta}}\,\dot{\boldsymbol{\eta}}\right)_i = \left(\frac{\partial T_{iq}}{\partial \eta_j}\,\frac{\partial \chi^{\boldsymbol{\alpha}}_q}{\partial \eta_k} + T_{iq}\,\frac{\partial^2 \chi^{\boldsymbol{\alpha}}_q}{\partial \eta_j\,\partial \eta_k}\right)\,\dot{\eta}_k \tag{5.111}$$

where

$$\frac{\partial T_{iq}}{\partial \eta_k} = \frac{\partial T_{iq}}{\partial \alpha_l}\,\frac{\partial \alpha_l}{\partial \eta_k} = \frac{1}{2}\,\epsilon_{iql}\,\frac{\partial \chi^{\boldsymbol{\alpha}}_l}{\partial \eta_k} \tag{5.112}$$

Introducing the antisymmetric third-order tensor $\mathbf{\Gamma^{\Omega}}$:

$$\Gamma^{\mathbf{\Omega}}_{ijk} = \frac{1}{2}\,\epsilon_{iql}\,\frac{\partial \chi^{\boldsymbol{\alpha}}_q}{\partial \eta_j}\,\frac{\partial \chi^{\boldsymbol{\alpha}}_l}{\partial \eta_k} \qquad \text{with} \qquad \Gamma^{\mathbf{\Omega}}_{ijk} = -\Gamma^{\mathbf{\Omega}}_{ikj} \tag{5.113}$$

we obtain

$$\left(\dot{\boldsymbol{\chi}}^{\mathbf{\Omega}\boldsymbol{\eta}}\right)_{ij} = \left(\Gamma^{\mathbf{\Omega}}_{ijk} + \Gamma^{\boldsymbol{\chi}^{\boldsymbol{\alpha}}}_{ijk}\right)\,\dot{\eta}_k \tag{5.114}$$

$$\frac{\partial}{\partial \eta_j}\left(\boldsymbol{\chi}^{\mathbf{\Omega}\boldsymbol{\eta}}\,\dot{\boldsymbol{\eta}}\right)_i = \left(\Gamma^{\mathbf{\Omega}}_{ikj} + \Gamma^{\boldsymbol{\chi}^{\boldsymbol{\alpha}}}_{ijk}\right)\,\dot{\eta}_k \tag{5.115}$$

Those expressions can be introduced in (5.109). Using the antisymmetry of $\mathbf{\Gamma^{\Omega}}$, and observing that $\Gamma^{\mathbf{\Omega}}_{ljk}\,\dot{\eta}_k\,\dot{\eta}_j = 0$, we get:

$$\left(\overline{\mathbf{g}}^{rot,A}_{iner}\right)_i = (\mathbf{J}\,\boldsymbol{\chi}^{\boldsymbol{\alpha}}_{\boldsymbol{\eta}})_{li}\,(\boldsymbol{\chi}^{\boldsymbol{\alpha}}_{\boldsymbol{\eta}}\,\ddot{\boldsymbol{\eta}})_l + \left[(\mathbf{J}\,\boldsymbol{\chi}^{\boldsymbol{\alpha}}_{\boldsymbol{\eta}})_{li}\,\Gamma^{\boldsymbol{\chi}^{\boldsymbol{\alpha}}}_{ljk} + 2\,\Gamma^{\mathbf{\Omega}}_{lik}\,(\mathbf{J}\,\boldsymbol{\chi}^{\boldsymbol{\alpha}}_{\boldsymbol{\eta}})_{lj}\right]\,\dot{\eta}_k\,\dot{\eta}_j \tag{5.116}$$

and in matrix form:

$$\overline{\mathbf{g}}^{rot,A}_{iner} = \boldsymbol{\chi}^{\boldsymbol{\alpha}T}_{\boldsymbol{\eta}}\,\mathbf{J}\,\boldsymbol{\chi}^{\boldsymbol{\alpha}}_{\boldsymbol{\eta}}\,\ddot{\boldsymbol{\eta}} + \boldsymbol{\chi}^{\boldsymbol{\alpha}T}_{\boldsymbol{\eta}}\,\mathbf{J}\,\left(\mathbf{\Gamma}^{\boldsymbol{\chi}^{\boldsymbol{\alpha}}}.\dot{\boldsymbol{\eta}}\right)\,\dot{\boldsymbol{\eta}} + 2\,\left(\mathbf{\Gamma}^{\mathbf{\Omega}}.\dot{\boldsymbol{\eta}}\right)^T\,\mathbf{J}\,\boldsymbol{\chi}^{\boldsymbol{\alpha}}_{\boldsymbol{\eta}}\,\dot{\boldsymbol{\eta}} \tag{5.117}$$

The *gyroscopic tensor of the rigid body* $\mathbf{h^{\Omega}}$ is defined by:

$$h^{\mathbf{\Omega}}_{ijk} = J_{jl}\,\epsilon_{lik} \tag{5.118}$$

After developments, the inertia forces can be restated:

$$\overline{\mathbf{g}}^{rot,A}_{iner} = \boldsymbol{\chi}^{\boldsymbol{\alpha}T}_{\boldsymbol{\eta}}\,\mathbf{J}\,\boldsymbol{\chi}^{\boldsymbol{\alpha}}_{\boldsymbol{\eta}}\,\ddot{\boldsymbol{\eta}} + \boldsymbol{\chi}^{\boldsymbol{\alpha}T}_{\boldsymbol{\eta}}\,\mathbf{J}\,\left(\mathbf{\Gamma}^{\boldsymbol{\chi}^{\boldsymbol{\alpha}}}.\dot{\boldsymbol{\eta}}\right)\,\dot{\boldsymbol{\eta}} + \boldsymbol{\chi}^{\boldsymbol{\alpha}T}_{\boldsymbol{\eta}}\,\left[\mathbf{h}^{\mathbf{\Omega}}.(\boldsymbol{\chi}^{\boldsymbol{\alpha}}_{\boldsymbol{\eta}}\,\dot{\boldsymbol{\eta}})\right]\,(\boldsymbol{\chi}^{\boldsymbol{\alpha}}_{\boldsymbol{\eta}}\,\dot{\boldsymbol{\eta}}) \tag{5.119}$$

This expression is composed of three terms:

- the inertia forces obtained when the variations of the reduced mass matrix are quasi-static,
- the equivalent forces associated with the curvature (or the nonlinearity) of the coordinate reduction formula,
- the third term is a gyroscopic force:

$$\overline{\mathbf{g}}^{gyr,A}_{iner} = \boldsymbol{\chi}^{\boldsymbol{\alpha}T}_{\boldsymbol{\eta}}\,\left[\mathbf{h}^{\mathbf{\Omega}}.\dot{\boldsymbol{\alpha}}\right]\,\dot{\boldsymbol{\alpha}} = \boldsymbol{\chi}^{\boldsymbol{\alpha}T}_{\boldsymbol{\eta}}\,\mathbf{g}^{gyr,A}_{iner} \tag{5.120}$$

which is interpreted as the projected gyroscopic forces of the initial model.

This generalized formulation for the inertia forces of an isolated rigid body is also valid in the particular case where no coordinate transformation is applied:

$$\boldsymbol{\eta} = \boldsymbol{\alpha}, \qquad \boldsymbol{\chi}_{\boldsymbol{\eta}}^{\boldsymbol{\alpha}} = \mathbf{I}, \qquad \boldsymbol{\Gamma}^{\chi^{\alpha}} = \mathbf{0} \tag{5.121}$$

Let us demonstrate the connection of equations (5.119) with the classic Euler equations in this case. For a rigid body described by a diagonal inertia tensor

$$\mathbf{J} = \begin{bmatrix} J_1 & 0 & 0 \\ 0 & J_2 & 0 \\ 0 & 0 & J_3 \end{bmatrix} \tag{5.122}$$

the non-zero components of the gyroscopic tensor $\mathbf{h}^{\boldsymbol{\Omega}}$ are:

$$\begin{aligned} h^{\boldsymbol{\Omega}}_{123} &= -J_2, \qquad h^{\boldsymbol{\Omega}}_{132} = J_3, \\ h^{\boldsymbol{\Omega}}_{213} &= J_1, \qquad h^{\boldsymbol{\Omega}}_{231} = -J_3, \\ h^{\boldsymbol{\Omega}}_{312} &= -J_1, \qquad h^{\boldsymbol{\Omega}}_{321} = J_2 \end{aligned} \tag{5.123}$$

Using $\boldsymbol{\eta} = \boldsymbol{\alpha}$, equation (5.119) leads to

$$\overline{\mathbf{g}}^{rot,A}_{iner} = \begin{bmatrix} J_1\ \ddot{\alpha}_1 + (J_3 - J_2)\ \dot{\alpha}_3\ \dot{\alpha}_2 \\ J_2\ \ddot{\alpha}_2 + (J_1 - J_3)\ \dot{\alpha}_1\ \dot{\alpha}_3 \\ J_3\ \ddot{\alpha}_3 + (J_2 - J_1)\ \dot{\alpha}_2\ \dot{\alpha}_1 \end{bmatrix} \tag{5.124}$$

The conclusion comes from the observation that $\dot{\boldsymbol{\alpha}} = \boldsymbol{\Omega}$ and $\ddot{\boldsymbol{\alpha}} = \dot{\boldsymbol{\Omega}}$ for an updated configuration. Indeed, we have:

$$\dot{T}_{ij}\ \dot{\alpha}_j = \frac{1}{2}\epsilon_{ijk}\ \dot{\alpha}_k\ \dot{\alpha}_j = 0 \qquad \Rightarrow \qquad \dot{\boldsymbol{\Omega}} = \mathbf{T}\ \ddot{\boldsymbol{\alpha}} + \dot{\mathbf{T}}\ \dot{\boldsymbol{\alpha}} = \ddot{\boldsymbol{\alpha}} \tag{5.125}$$

Inertia forces associated with the global rotation kinetic energy

The inertia forces associated with the rotation dofs $\boldsymbol{\psi}$ involve the contributions of all rigid-bodies of the mechanism (see previous section), as well as the slightly more complex contributions of the flexible bodies. However, it is easily verified that the global forces can still be decomposed into three terms:

- a force proportional to the modal accelerations:

$$\boldsymbol{\chi}_{\boldsymbol{\eta}}^{\boldsymbol{\psi}T}\ \mathbf{M}^{\boldsymbol{\psi}\boldsymbol{\psi}}\ \boldsymbol{\chi}_{\boldsymbol{\eta}}^{\boldsymbol{\psi}}\ \ddot{\boldsymbol{\eta}} \tag{5.126}$$

- an equivalent force associated with the curvature tensor of the coordinate transformation:

$$\boldsymbol{\chi}_{\boldsymbol{\eta}}^{\psi T}\ \mathbf{M}^{\psi\psi}\ \left(\boldsymbol{\Gamma}^{\chi^{\psi}}.\dot{\boldsymbol{\eta}}\right)\ \dot{\boldsymbol{\eta}} \tag{5.127}$$

- the projected gyroscopic forces

$$\boldsymbol{\chi}_{\boldsymbol{\eta}}^{\psi T}\ \left[\mathbf{h}^{gyr}.(\boldsymbol{\chi}_{\boldsymbol{\eta}}^{\psi}\ \dot{\boldsymbol{\eta}})\right]\ (\boldsymbol{\chi}_{\boldsymbol{\eta}}^{\psi}\ \dot{\boldsymbol{\eta}}) = \boldsymbol{\chi}_{\boldsymbol{\eta}}^{\psi T}\ \left(\mathbf{h}^{gyr}.\dot{\boldsymbol{\psi}}\right)\ \dot{\boldsymbol{\psi}} \tag{5.128}$$

where $\mathbf{h}^{gyr}$ is the gyroscopic tensor of the initial model.

Assembled inertia forces

As a conclusion of the previous developments, the assembled inertia forces have the form

$$\overline{\mathbf{g}}_{iner} = \overline{\mathbf{M}}\ \ddot{\boldsymbol{\eta}} + \left(\overline{\mathbf{h}}.\dot{\boldsymbol{\eta}}\right)\ \dot{\boldsymbol{\eta}} \tag{5.129}$$

with

$$\overline{\mathbf{M}} = \boldsymbol{\chi}_{\boldsymbol{\eta}}^{T}\ \mathbf{M}\ \boldsymbol{\chi}_{\boldsymbol{\eta}} \tag{5.130}$$

$$\overline{\mathbf{h}} = \overline{\mathbf{h}}^{\boldsymbol{\chi}} + \overline{\mathbf{h}}^{gyr} \tag{5.131}$$

The expression of the reduced mass matrix is classical, and two different third-order tensors are responsible for quadratic forces in the modal rates.

(i) The tensor $\overline{\mathbf{h}}^{\boldsymbol{\chi}}$ is associated with the curvature $\boldsymbol{\Gamma}^{\boldsymbol{\chi}}$ of the reduced parameterization:

$$\overline{h}_{ijk}^{\boldsymbol{\chi}} = \left(\boldsymbol{\chi}_{\boldsymbol{\eta}}^{T}\ \mathbf{M}\right)_{il}\ \Gamma_{ljk}^{\boldsymbol{\chi}} \tag{5.132}$$

$\boldsymbol{\Gamma}^{\boldsymbol{\chi}}$ can be estimated by differentiation of the Jacobian $\boldsymbol{\chi}_{\boldsymbol{\eta}}$ using a finite difference approach.

(ii) The tensor $\overline{\mathbf{h}}^{gyr}$ satisfies

$$(\overline{\mathbf{h}}^{gyr}.\dot{\boldsymbol{\eta}})\ \dot{\boldsymbol{\eta}} = \boldsymbol{\chi}_{\boldsymbol{\eta}}^{\psi T}\ \left[\mathbf{h}^{gyr}.(\boldsymbol{\chi}_{\boldsymbol{\eta}}^{\psi}\ \dot{\boldsymbol{\eta}})\right]\ (\boldsymbol{\chi}_{\boldsymbol{\eta}}^{\psi}\ \dot{\boldsymbol{\eta}}) = \boldsymbol{\chi}_{\boldsymbol{\eta}}^{\psi T}\ \mathbf{g}^{gyr} \tag{5.133}$$

and it could be computed from $\mathbf{h}^{gyr}$, according to a tensor projection. However, in the Finite Element code, manipulations of third-order tensors are avoided and $\mathbf{h}^{gyr}$ is not available. For this reason, we propose to proceed by inspection from a computation of the gyroscopic force vector $\mathbf{g}^{gyr}$. A unit velocity is imposed to each pair of variables $(\dot{\eta}_i, \dot{\eta}_j)$, and we deduce the rates of the Finite Element coordinates $\dot{\mathbf{q}} = \boldsymbol{\chi}_{\boldsymbol{\eta}}\ \dot{\boldsymbol{\eta}}$. Then, the gyroscopic forces $\mathbf{g}^{gyr}(\mathbf{q}, \dot{\mathbf{q}})$ are obtained using standard Finite Element routines. This vector is projected to obtain the equivalent

gyroscopic forces $\overline{\mathbf{g}}^{gyr} = \boldsymbol{\chi}_{\boldsymbol{\eta}}^T \, \mathbf{g}^{gyr}$, which allows a component-wise identification of the gyroscopic tensor $\overline{\mathbf{h}}^{gyr}$.

Simplification of the inertia forces

Theoretically, the mass matrix and the gyroscopic tensor formulated above depend on the whole set of reduced parameters $\boldsymbol{\eta}$. However, in the following, the dependence with respect to the flexible coordinates $\boldsymbol{\eta}^\delta$ is neglected, so that we have

$$\overline{\mathbf{g}}_{iner} = \overline{\mathbf{M}}(\boldsymbol{\theta}) \; \ddot{\boldsymbol{\eta}} + \left(\overline{\mathbf{h}}(\boldsymbol{\theta}).\dot{\boldsymbol{\eta}}\right) \; \dot{\boldsymbol{\eta}} \tag{5.134}$$

The mass matrix and the gyroscopic tensor are only evaluated at undeformed configurations.

The curvature tensor is required for the construction of $\overline{\mathbf{h}}$. From the expression of the Jacobian (5.61), it has an interesting structure at an undeformed configuration:

$$\frac{\partial^2 \chi_i}{\partial \theta_j \partial \theta_k} = \frac{\partial^2 \rho_i}{\partial \theta_j \partial \theta_k} = \frac{\partial \Psi_{ij}^{\mathbf{u}\boldsymbol{\theta}}}{\partial \theta_k} \tag{5.135}$$

$$\frac{\partial^2 \chi_i}{\partial \theta_j \partial \eta_k^\delta} = \frac{\partial \Psi_{ik}^{\mathbf{u}\delta}}{\partial \theta_j} \tag{5.136}$$

$$\frac{\partial^2 \chi_i}{\partial \eta_k^\delta \partial \eta_j^\delta} = 0 \tag{5.137}$$

It is associated with the sensitivities of the rigid and flexible modes, which are easily estimated by finite difference.

5.5.3 Reduced equations of motion

After the development of the elastic and inertia forces, the equivalent external loads are estimated from a virtual work expression:

$$\delta \mathcal{W} = \mathbf{g}_{ext}^T \, \delta \mathbf{q} = \mathbf{g}_{ext}^T \; \boldsymbol{\Psi}^{\mathbf{q}\boldsymbol{\eta}} \; \delta \boldsymbol{\eta} = \overline{\mathbf{g}}_{ext}^T \, \delta \boldsymbol{\eta}, \qquad \text{with} \qquad \overline{\mathbf{g}}_{ext} = \boldsymbol{\Psi}^{\mathbf{q}\boldsymbol{\eta}T} \; \mathbf{g}_{ext} \tag{5.138}$$

The *reduced equations of motion* follow

$$\boxed{\overline{\mathbf{M}}(\boldsymbol{\theta}) \; \ddot{\boldsymbol{\eta}} + \left(\overline{\mathbf{h}}(\boldsymbol{\theta}).\dot{\boldsymbol{\eta}}\right) \; \dot{\boldsymbol{\eta}} + \overline{\mathbf{K}}(\boldsymbol{\theta}) \; \boldsymbol{\eta} = \overline{\mathbf{g}}_{ext}} \tag{5.139}$$

with the structure:

$$\overline{\mathbf{M}} = \begin{bmatrix} \overline{\mathbf{M}}^{\boldsymbol{\theta}\boldsymbol{\theta}} & \overline{\mathbf{M}}^{\boldsymbol{\theta}\gamma} & \overline{\mathbf{M}}^{\boldsymbol{\theta}\iota} \\ \overline{\mathbf{M}}^{\gamma\boldsymbol{\theta}} & \overline{\mathbf{M}}^{\gamma\gamma} & \overline{\mathbf{M}}^{\gamma\iota} \\ \overline{\mathbf{M}}^{\iota\boldsymbol{\theta}} & \overline{\mathbf{M}}^{\iota\gamma} & \mathbf{I} \end{bmatrix}, \quad \overline{\mathbf{K}} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \overline{\mathbf{K}}^{\gamma\gamma} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Omega}^2 \end{bmatrix}, \quad \overline{\mathbf{g}}_{ext} = \begin{bmatrix} \mathbf{g}^a_{ext} + \boldsymbol{\Psi}^{g\boldsymbol{\theta}T}\, \mathbf{g}^g_{ext} \\ \mathbf{g}^g_{ext} \\ \mathbf{0} \end{bmatrix} \tag{5.140}$$

$\mathbf{g}^a_{ext}$ denotes the actuator forces, and $\mathbf{g}^g_{ext}$ the forces applied to the constraint dofs. This model is fully described by $\overline{\mathbf{M}}$, $\overline{\mathbf{K}}$, $\overline{\mathbf{h}}$ and $\boldsymbol{\Psi}^{g\boldsymbol{\theta}}$, which smoothly depend on $\boldsymbol{\theta}$.

5.5.4 Algorithm for local model reduction

Assuming that a reference model is available at a configuration $\boldsymbol{\theta}_{ref}$ (with local modes $\boldsymbol{\Psi}_{ref}$), the construction of the reduced-order model for a configuration $\boldsymbol{\theta}$ involves the following steps:

1. kinematic analysis (from $\boldsymbol{\theta}_{ref}$ to $\boldsymbol{\theta}$),
2. local mode synthesis ($\boldsymbol{\Psi}^{\mathbf{u}\boldsymbol{\eta}}$),
3. mode matching ($\boldsymbol{\Psi} - \boldsymbol{\Psi}_{ref}$),
4. model reduction ($\overline{\mathbf{M}}$, $\overline{\mathbf{K}}$, $\overline{\mathbf{h}}$ and $\boldsymbol{\Psi}^{g\boldsymbol{\theta}}$).

An overview of the local order reduction algorithm is given in Figure 5.9. This local analysis is combined with a configuration space approximation algorithm, presented in the next section. Hence, a database is exploited to store and reuse the local models. As mentioned earlier, the mode matching may fail if the reference configuration $\boldsymbol{\theta}_{ref}$ is too far from $\boldsymbol{\theta}$, leading to a non-valid model.

5.6 Approximation in the configuration space

The local construction of the reduced-order model involves a computationally demanding numerical procedure. However, the variations of the model in the configuration space can be described by a simplified and approximated metamodel, or model of the model. If all relevant coefficients associated with $\overline{\mathbf{M}}$, $\overline{\mathbf{K}}$, $\overline{\mathbf{h}}$ and $\boldsymbol{\Psi}^{g\boldsymbol{\theta}}$ are collected in a single $t \times 1$ output vector $\mathbf{f}$, the numerical reduction procedure

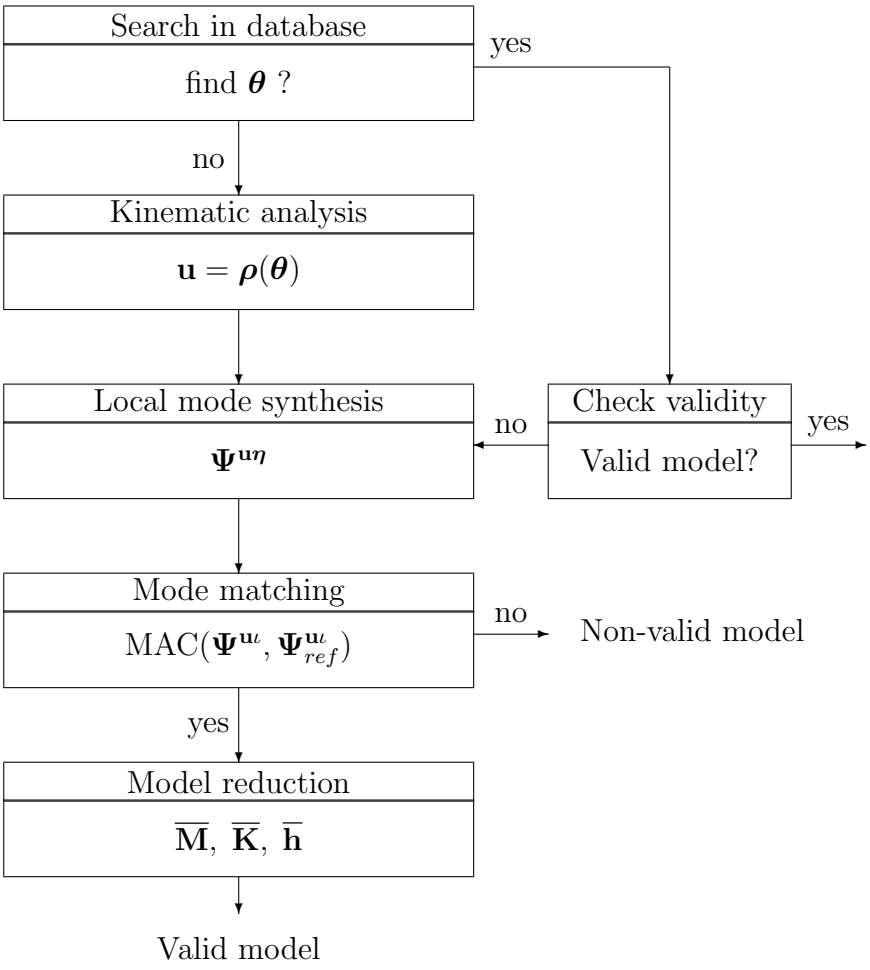


Figure 5.9: Construction of the reduced model in the neighborhood of a configuration $\boldsymbol{\theta}$.

is a smooth black box function $\mathbf{f}(\boldsymbol{\theta})$, and the metamodel is an approximation $\widehat{\mathbf{f}}(\boldsymbol{\theta})$ for this function:

$$\mathbf{f} = \mathbf{f}(\boldsymbol{\theta}) \qquad \Rightarrow \qquad \mathbf{f} \simeq \widehat{\mathbf{f}}(\boldsymbol{\theta}) \qquad \forall\, \boldsymbol{\theta} \in \Omega_{\boldsymbol{\theta}}^{r} \tag{5.141}$$

Obviously, there is a trade-off between the accuracy of the approximation and the complexity of the metamodel.

This standard approximation problem received a major interest in the literature. In particular, a strong connection can be established with response surface methods proposed in optimization [MM95], which take advantage of an approximated model to reduce the number of runs of a full model.

In the state of the art (section 2.4.2), several approximation techniques were reported, such as neural networks, polynomial approximations, rational approximations, kriging functions and the local model network approach. In general, the metamodel is elaborated according to the following procedure:

1. select a set of inputs $\boldsymbol{\theta}^{(1)}, ..., \boldsymbol{\theta}^{(r)}$;
2. run the local reduction algorithm for each input and get back $\mathbf{f}^{(1)} ... \mathbf{f}^{(r)}$;
3. define a generic approximated function $\widehat{\mathbf{f}}(\mathbf{W}, \boldsymbol{\theta})$, where $\mathbf{W}$ is a matrix of free parameters;
4. select $\mathbf{W}$ to fit $\widehat{\mathbf{f}}$ on the data obtained in step 2.

Step 1 is also referred to as the experimental design problem. Step 2 was the subject of our previous developments. In step 3, many choices are possible, and we restrict the study to linear approximations:

$$\widehat{f}_i(\boldsymbol{\theta}) = \sum_{j=1}^{n_v} W_{ij}\; v_j(\boldsymbol{\theta}) \tag{5.142}$$

where v_j $(j = 1, ..., n_v)$ are fixed basis functions, and W_{ij} are the free weights which allow to fit the model. The linearity is with respect to the weights, and not to the input variables. Step 4 is then a linear regression problem which can be solved in the least-square sense using standard methods.

Radial basis functions are an interesting candidate for the basis functions:

$$v_j = v_j(\|\boldsymbol{\theta} - \boldsymbol{\theta}^{(j)}\|) \qquad j = 1, ..., r \tag{5.143}$$

If the number of radial functions equals the number of given configurations $\boldsymbol{\theta}^{(j)}$, it is possible to interpolate exactly any set of scattered data. However, for large data sets, the resulting function $\widehat{\mathbf{f}}$ may become computationally inefficient, and the number of basis functions should be reduced following a non-trivial selection procedure. Moreover, the choice of appropriate radial functions is not systematic, and sometimes requires nonlinear optimization strategies.

In this research, we prefer low-order polynomial basis functions, which are more simple and efficient for our problem. In order to increase the flexibility of the approximation, a piecewise strategy is adopted as proposed in [BDG06], so that the above procedure is applied for several non-overlapping subsets of the configuration space $\Omega^r_{\boldsymbol{\theta}}$. The decomposition into subsets can be realized adaptively in order to satisfy a specification on the approximation error. Thus, a general, efficient and systematic approximation procedure is developed leading to a portable and computationally efficient nonlinear model.

The next section presents the piecewise strategy for approximation. Afterwards, the polynomial basis functions are defined and an algorithm is proposed for automatic configuration space decomposition.

5.6.1 Piecewise strategy

A piecewise approximation is defined over a collection of subsets that cover the configuration space. In computational mechanics, several methodologies have been proposed to enforce the consistency of a piecewise approximation, *e.g.* in a Finite Element context. Most of them are dedicated to 2 or 3 dimensional spaces, and they are not applicable for our problem if the configuration space has a higher dimension. This section describes a more general approach for the decomposition of the configuration space using the concept of subpaving, developed in the theory of interval analysis.

Interval analysis theory [JKDW01] offers systematic methods for the approximation of a complex set using simple subsets. If $[\theta] \subset \mathcal{R}$ represents a real interval, with a lower and an upper bounds:

$$[\theta] = [\underline{\theta} \quad \overline{\theta}] \tag{5.144}$$

an interval vector $[\boldsymbol{\theta}] \subset \mathcal{R}^s$ is defined by the cartesian product of s real inter-

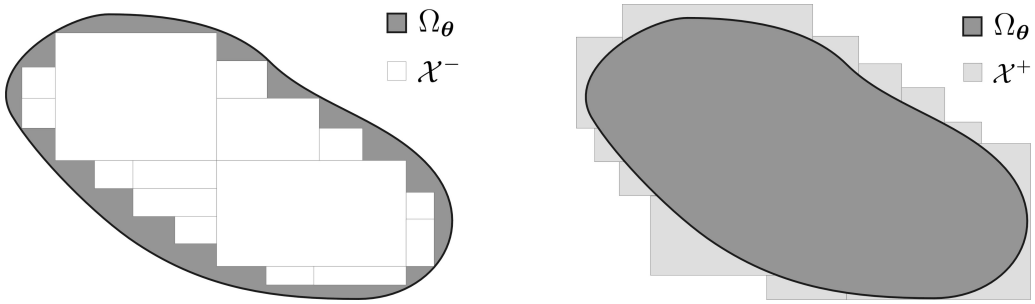


Figure 5.10: Inner and outer approximation of the configuration space using subpavings.

vals $[\theta_i]$:

$$[\boldsymbol{\theta}] = [\theta_1] \times [\theta_2] \times \ldots \times [\theta_s] \tag{5.145}$$

Roughly speaking, an interval vector is a s-dimensional box that can be represented by two corners: the lower bound $\underline{\boldsymbol{\theta}} = [\underline{\theta_1} \quad \underline{\theta_2} \ldots \underline{\theta_s}]^T$ and the upper bound $\overline{\boldsymbol{\theta}} = [\overline{\theta_1} \quad \overline{\theta_2} \ldots \overline{\theta_s}]^T$.

A subpaving $\mathcal{X}$ is a set of n_b non-overlapping boxes[4]:

$$\mathcal{X} = \{[\boldsymbol{\theta}]^1, \ldots, [\boldsymbol{\theta}]^{n_b}\} \tag{5.146}$$

The configuration space $\Omega^r_{\boldsymbol{\theta}}$ can be approximated from inside and outside by two subpavings, see Figure 5.10:

$$\mathcal{X}^- \subset \Omega^r_{\boldsymbol{\theta}} \subset \mathcal{X}^+ \tag{5.147}$$

A subpaving is an appropriate basis for the definition of piecewise functions. If $\mathcal{B} = \{0, 1\}$ denotes the set of booleans, we define the activation functions $\tau^k : \mathcal{X} \to \mathcal{B}$:

$$\begin{array}{ll} \tau^k(\boldsymbol{\theta}) = 1 & \forall\ \boldsymbol{\theta} \in [\boldsymbol{\theta}]^k \\ \tau^k(\boldsymbol{\theta}) = 0 & \forall\ \boldsymbol{\theta} \notin [\boldsymbol{\theta}]^k \end{array} \qquad k = 1, ..., n_b \tag{5.148}$$

and a piecewise approximation function can be formulated

$$\widehat{f_i}(\boldsymbol{\theta}) = \sum_{k=1}^{n_b} \sum_{j=1}^{n_v} W_{ij}^k \ \tau^k(\boldsymbol{\theta}) \ v_j(\boldsymbol{\theta} - \underline{\boldsymbol{\theta}}^k) \tag{5.149}$$

[4]The interested reader may find a more accurate definition in [JKDW01].

where v_j are a low-order polynomials, defined for each boxes in local coordinates. Hence, the construction of an approximated piecewise model in the configuration space involves two problems:

- the definition of an appropriate inner subpaving $\mathcal{X}^- \subset \Omega^r_{\boldsymbol{\theta}}$,
- the selection of local basis functions v_j.

In the following, possible choices for polynomial approximation are first discussed; the construction of an appropriate subpaving is addressed later.

5.6.2 Local polynomial approximation

Two approaches for the local polynomial approximation are hereby considered and compared: quadratic polynomials and Lagrange polynomials. For notational convenience, the presentation focuses on a single component i of the model $f = \mathbf{f}_i$ which depends on a s-dimensional configuration vector $\boldsymbol{\theta}$. The interpolating polynomial is denoted $P_s = \widehat{\mathbf{f}}_i$, and the weight matrix becomes a vector $\mathbf{w}_j = W_{ij}$.

Quadratic polynomials

For a one-dimensional problem, a quadratic polynomial P_1 is defined by

$$P_1 = a + b\,\theta + c\,\theta^2 \tag{5.150}$$

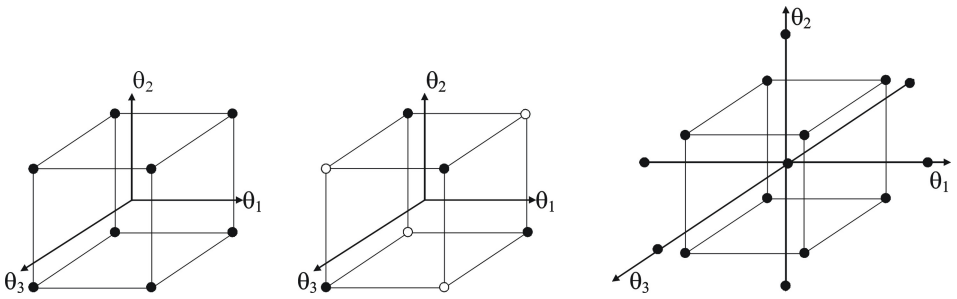
This formula can be generalized to s-dimensional problems:

$$P_s = a + \sum_{i=1}^{s} b_i\,\theta_i + \sum_{i=1}^{s}\sum_{j=i}^{s} c_{ij}\,\theta_i\,\theta_j \tag{5.151}$$

The basis functions v_k ($k = 1, ..., n_v$) are the monomials 1, θ_i ($i = 1, ..., s$) and $\theta_i\,\theta_j$ ($i = 1, ..., s$, $j = i, ..., s$), and their weights are the coefficients of the polynomial. If one assumes that the products $\theta_i\,\theta_j$ are pre-computed[5], the computation of P_s requires n_{op} floating-point operations, with:

$$n_{op}(s) = s^2 + 3s \tag{5.152}$$

[5]For a function $\mathbf{f} : \mathcal{R}^s \to \mathcal{R}^t$, the values of $\theta_i\,\theta_j$ are computed once for the t output components, and the associated computational cost is thus far less significant than for the other operations.



Full factorial Fractional Factorial Central Composite Design

Figure 5.11: Experimental designs in 3D [Mon97].

Considering a set of r data $f^{(k)} = f(\theta^{(k)})$, $(k = 1, ..., r$ and $r > n_v)$, the definition of the weights is a linear regression problem:

$$\begin{bmatrix} f^{(1)} \\ \vdots \\ f^{(r)} \end{bmatrix} = \begin{bmatrix} v_1^{(1)} & \dots & v_{n_v}^{(1)} \\ \vdots & \ddots & \vdots \\ v_1^{(r)} & \dots & v_{n_v}^{(r)} \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_{n_v} \end{bmatrix} \tag{5.153}$$

where $v_j^{(k)} = v_j(\boldsymbol{\theta}^{(k)})$. This overdetermined set of equations can be solved in the least square sense, using standard regression algorithms (normal equations, QR algorithm or Singular Value Decomposition).

Several authors (*e.g.* Montgomery [Mon97]) argued that the data points $\boldsymbol{\theta}^{(k)}$ should be selected according to an experimental design in order to improve the quality of the approximation. Figure 5.11 illustrates classical definitions: the full factorial design, the fractional factorial design, and the central composite design. More sophisticated methods, such as the Taguchi approach, seem unnecessarily complicated for our problem.

The central composite design is retained here, since it provides sufficient information to estimate the quadratic effects required for a second-order polynomial approximation. The axial position of the "star points" is selected at the intersection with the border of the box.[6]

Due to the piecewise strategy, the approximation function exhibits a discontinuous behavior at every boundary between boxes. This important drawback can be overcome using the family of Lagrange polynomials (see [ZT89, PTVF92]

[6]For one-dimensional problems, in order to avoid superimposation on the other data points, the star points are fixed at intermediate positions.

for detailed presentations), so that a C_0 continuity is obtained at every boundary between matching boxes.

Lagrange polynomials

Let us consider the approximation of a *real function* $f : \mathcal{R} \to \mathcal{R}$, from p data: $f^{(k)} = f(\theta^{(k)})$, $k = 1, ..., p$. It is well-known that an order $p-1$ polynomial can be fitted exactly on those data, for instance using the p Lagrange polynomials $L^{(i)}$, which are constructed to satisfy the condition:

$$L^{(i)}(\theta^{(k)}) = \delta_{ik} \tag{5.154}$$

δ_{ik} is the kronecker symbol. This condition leads to the Lagrange interpolation formula:

$$L^{(i)}(\theta) = \prod_{\substack{k=1 \\ (k \neq i)}}^{p} \frac{\theta - \theta^{(k)}}{\theta^{(i)} - \theta^{(k)}} = \sum_{k=1}^{p} A_{ik}\ \theta^{k-1} \tag{5.155}$$

The last equality defines the components of the matrix $\mathbf{A}$, which is implemented in the numerical code for efficient computations. Therefore, the interpolating polynomial of the function f is simply

$$P_1(\theta) = \sum_{i=1}^{p} f^{(i)}\ L^{(i)}(\theta) = \sum_{k=1}^{p} f^{*(k)}\ \theta^k \tag{5.156}$$

with

$$f^{*(k)} = \sum_{i=1}^{p} f^{(i)}\ A_{ik} \tag{5.157}$$

This idea can be recursively implemented for *s-dimensional problems*, if the data points are placed in a s-dimensional grid. Suppose that an interpolating polynomial P_{s-1} is available for $s-1$ dimensional problems, and consider p_s values for the additional coordinate θ_s. The s-dimensional interpolating polynomial is obtained in two steps:

- for each fixed value $\theta_s^{(i)}$ $(i = 1, ..., p_s)$, define an interpolating polynomial $P_{s-1}^{(i)}$ for the $s-1$ dimensional problem,
- exploit 1-dimensional polynomials $L^{(i)}$ $(i = 1, ..., p_s)$ for interpolation in direction θ_s.

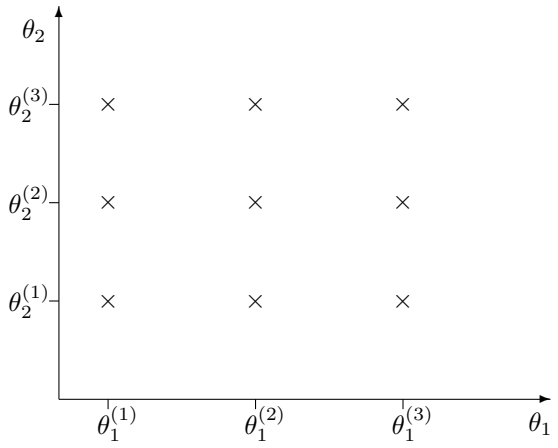


Figure 5.12: Two-dimensional configuration space with a quadratic interpolation formula ($s = 2$, $p = 3$).

This is summarized in the following formula:

$$P_s(\boldsymbol{\theta}) = \sum_{i=1}^{p_s} P_{s-1}^{(i)} \; L^{(i)}(\theta_s) = \sum_{k=1}^{p_s} P_{s-1}^{*(k)} \; \theta_s^k \tag{5.158}$$

with

$$P_{s-1}^{*(k)} = \sum_{i=1}^{p} \; P_{s-1}^{(i)} \; A_{ik} \tag{5.159}$$

According to this last equation, the construction of $P_{s-1}^{*(k)}$ relies on a recursive implementation of addition and scalar multiplication for polynomials. In Figure 5.12, a two-dimensional problem with a quadratic interpolation formula ($s = 2$, $p = 3$) is considered, and equation (5.158) becomes

$$P_2(\theta_1, \theta_2) = \sum_{i=1}^{3} \sum_{j=1}^{3} \; f\left(\theta_1^{(i)}, \theta_2^{(j)}\right) \; L_i(\theta_1) \; L_j(\theta_2) \tag{5.160}$$

A remarkable properties of the Lagrange interpolation, is that the $s-1$ dimensional function obtained by fixing $\theta_s = \theta_s^{(k)}$ is not influenced by the data associated with $\theta_s^{(i)}$, $i \neq k$. Therefore, if two neighbor boxes share the same $s-1$ dimensional grid at their boundary, *the piecewise interpolation is continuous.* The continuity is lost when the subpaving is not made of matching boxes, for instance due to an adaptive strategy for the configuration space decomposition, as illustrated in Figure 5.13. Such an adaptive strategy is usually adopted to improve the compromise between accuracy and memory storage. Hence, the continuity of

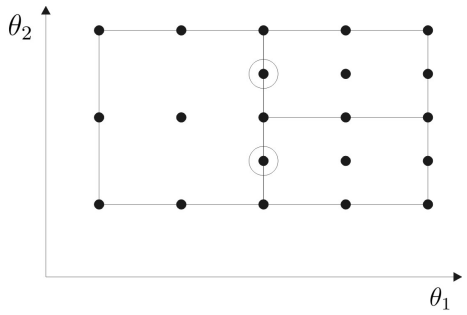


Figure 5.13: Discontinuity of Lagrange polynomials between boxes with non-matching grids: the circled data points do not belong to the grid of the left box.

the model is a third conflicting criterion to be considered for the definition of an appropriate approximation strategy.

In this work, the second-order Lagrange interpolation has been selected, which relies on $p = 3$ data in the 1-dimensional case:

$$P_1 = f^{*(1)} + f^{*(2)}\,\theta + f^{*(3)}\,\theta^2 \tag{5.161}$$

If θ^2 is pre-computed[7], each component of $\widehat{\mathbf{f}}$ requires $n_{op}(1) = 4$ operations (2 additions and 2 multiplications).

An s-dimensional interpolation involves 3^{s-1} 1-dimensional polynomials, and we get:

$$P_s(\boldsymbol{\theta}) = P_{s-1}^{*(1)} + P_{s-1}^{*(2)}\,\theta_s + P_{s-1}^{*(3)}\,\theta_s^2 \tag{5.162}$$

the number of operations per component n_{op} satisfies

$$n_{op}(s) = 3\,n_{op}(s-1) + 4 \tag{5.163}$$

This difference equations has the solution

$$n_{op}(s) = 2\,(3^s - 1) \tag{5.164}$$

In order to make a connection with standard approximation techniques, equation (5.162) could be rewritten as a sum of 3^s monoms or basis functions, with 3^s weights allowing to fit exactly the 3^s data.

[7]As for quadratic polynomials, the computation of θ^2 is computed once for the t output components, and the associated computational cost is far less significant than for the other operations.

s	n_{op}		n_{db}		n_{cfg}	
	Quadratic	Lagrange	Quadratic	Lagrange	Quadratic	Lagrange
1	4	4	3	3	5	3
2	10	16	6	9	9	9
3	18	52	10	27	15	27
4	28	160	15	81	25	81
5	40	484	21	243	43	243
6	54	1456	28	729	77	729
s	s^2+3s	$2(3^s-1)$	$\frac{s^2+3s+2}{2}$	3^s	2^s+2s+1	3^s

Table 5.1: Comparison between quadratic and Lagrange polynomials.

Comparison Quadratic/Lagrange polynomials

The Lagrange polynomials have more flexibility, with the advantages of exact interpolation, and C_0 continuity for neighboring boxes with matching grids. Moreover, recursive concepts make the implementation reliable and efficient. The price to pay is the manipulation of higher-order polynomials, whose construction requires more data points. For dimensions $s = 1$ to 6, Table 5.1 compares quadratic and Lagrange approximation for a function $f : \mathcal{R}^s \to \mathcal{R}$ with respect to

- the number of operations n_{op} to estimate P_s,
- the number of weights n_{db}, that should be eventually stored in a database for each component of the model,
- the number of data points n_{cfg} required to compute the weights.

From this analysis, Lagrange polynomials are less attractive for high-dimensional problems. Anyway, a compromise has to be defined between:

- the accuracy,
- the continuity,
- the computational load for the approximation function,
- the memory storage requirement,

- the number of data point necessary to construct the approximation.

5.6.3 Automatic configuration space decomposition

Quadratic and Lagrange polynomials are defined in a box $[\boldsymbol{\theta}]$, on the basis of information collected at the data points. The approximation error within the box is directly connected with

- the smoothness of $\mathbf{f}$ in $[\boldsymbol{\theta}]$,
- the size of $[\boldsymbol{\theta}]$.

Ideally, the size of the boxes should be adapted according to the behavior of $\mathbf{f}$, in order to improve the compromise between accuracy, continuity, memory storage and efficient construction of the approximation function. For instance, close to singularities, a refinement is suitable to track the strong variations of the dynamic parameters. The adaptive algorithm presented here relies on the representation of the subpaving as *tree data structure* and it allows a high-level definition of the approximation error.

Before the description of the algorithm, let us define the algorithmic object `model`, at the basis of the configuration space inspection procedure. A `model` is a data structure containing:

- the box $[\boldsymbol{\theta}]$, defined by $\underline{\boldsymbol{\theta}}$ and $\overline{\boldsymbol{\theta}}$,
- a pointer to the approximation function $\widehat{\mathbf{f}}$,
- a boolean `valid`, true if the approximated function is a valid model,
- two pointers `leftModel`, and `rightModel` to child models that might be defined after the bisection of the box $[\boldsymbol{\theta}]$, thus, a `model` is binary tree structure which represents a subpaving,
- an integer `bisectDirection`, that specifies the possible bisected dimension.

Temporary variables are also defined

- e^{val} and $\widehat{e}_i$ $(i = 1, ..., s)$ are representative of the approximation error in the box, they will be defined later,

- `cfgSet` is the set of configurations defined in the box according to the experimental design,

- $\boldsymbol{\theta}_{ref}$ is a reference configuration extracted from the database, for which a valid local model is available; it may belong to `cfgSet`, otherwise it should be as close as possible to the box $[\boldsymbol{\theta}]$,

- $\boldsymbol{\theta}$ is the current configuration.

Figure 5.14 illustrates the construction of a `model` associated with a box $[\boldsymbol{\theta}]$. Local models are defined for every data points of the configuration set `cfgSet`, and after, the approximation function is constructed. The `model` is bisected if the local analysis failed at any data point or if the approximation error is not acceptable. In those cases, a bisection of the model leads to the recursive definition of two child models. In the following, the error analysis, and the selection of the bisection direction are presented with more details.

Error analysis

The *relative approximation error* is defined by:

$$e(\boldsymbol{\theta}) = \frac{\left\| \mathbf{f}(\boldsymbol{\theta}) - \widehat{\mathbf{f}}(\boldsymbol{\theta}) \right\|}{\|\mathbf{f}(\boldsymbol{\theta})\|} > 0 \tag{5.165}$$

Since the components of the vector $\mathbf{f}$ do not have the same physical meaning and the same order of magnitude, it is recommended to replace the norm by a weighted norm:

$$\|\mathbf{f}\|_{\mathbf{D}} = \sqrt{\mathbf{f}^T \, \mathbf{D} \, \mathbf{f}} \tag{5.166}$$

where $\mathbf{D} = diag(d_1, ..., d_t)$ is a diagonal matrix of positive weights. Each weight d_i should be adjusted according to the physical meaning of the component f_i (*e.g.* component of $\overline{\mathbf{M}}^{\boldsymbol{\theta}\boldsymbol{\theta}}$, component of $\overline{\mathbf{K}}^{\gamma\gamma}$, etc.).

The validation of the approximation relies on the analysis of the error for a set of validation configurations $\boldsymbol{\theta}^{val(i)}$ $(i = 1, ..., n_{val})$. The average error of the validation set is:

$$e^{val} = \frac{1}{n_{val}} \sum_{i=1}^{n_{val}} e\left(\boldsymbol{\theta}^{val(i)}\right) \tag{5.167}$$

The *criterion for the validation of the approximation* is defined by

$$e^{val} < tol^e \tag{5.168}$$

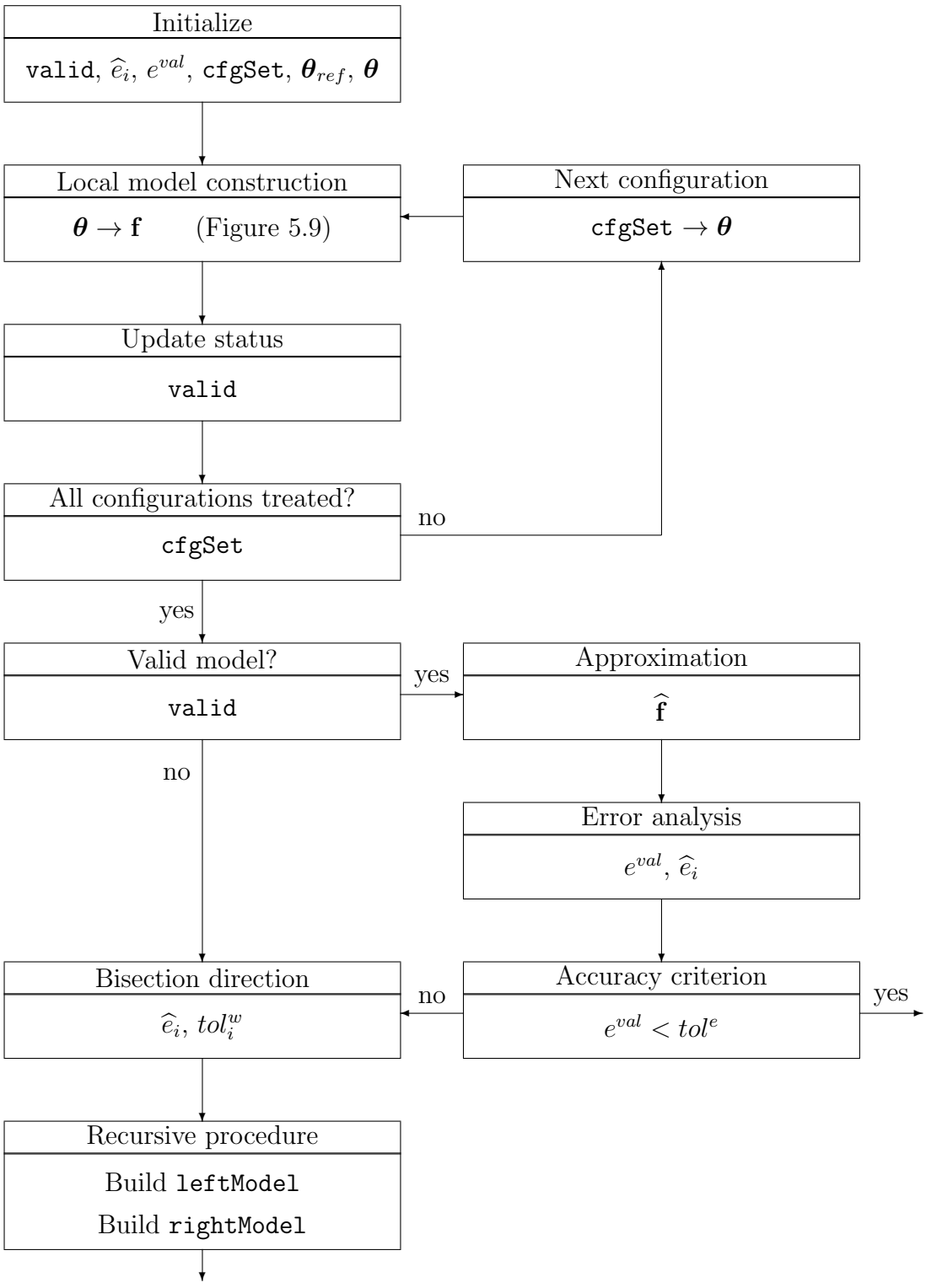


Figure 5.14: Construction of the global model in a box [$\boldsymbol{\theta}$].

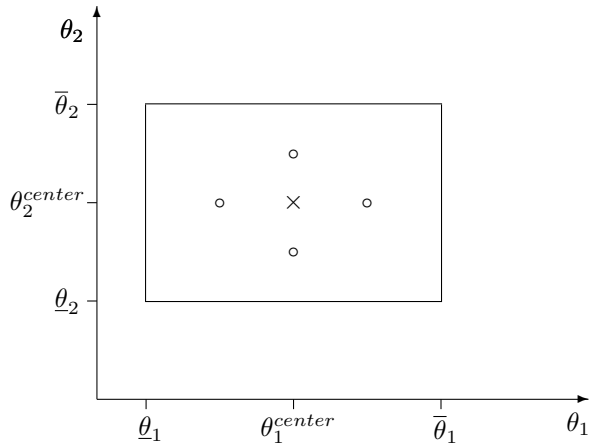


Figure 5.15: Validation configurations for error and sensitivity analysis in a two dimensional configuration space. The validation points are represented by the circles.

where tol^e is a tolerance. For critical applications, e^{val} can be defined as the maximal error observed in the validation set.

In order to select the optimal bisection direction, the sensitivity of the relative error with respect to the configuration parameters $\boldsymbol{\theta}$ is analyzed. In particular, the sensitivity of e with respect to a component θ_i is defined by

$$\frac{\partial e}{\partial \theta_i} \tag{5.169}$$

Assuming that the error vanishes at the center of the box, a first order approximation of the maximal error encountered when moving to the side of the box along direction i is given by:

$$\widehat{e}_i = \left| \frac{\partial e}{\partial \theta_i} \left(\frac{\overline{\theta}_i - \underline{\theta}_i}{2} \right) \right| \tag{5.170}$$

Therefore, *the optimal bisection direction is defined as the direction which maximizes* $\widehat{e}_i$.

The sensitivities of the error are computed using a finite difference method. The validation set can be exploited for this purpose, for instance, by the definition of validation configurations at mid-distance star-points, as illustrated in Figure 5.15.

Selection of the bisection direction

As described in Figure 5.14, if the definition of a local model failed, or if the approximation error e^{val} is not acceptable, the algorithm achieves a bisection of the box into two child boxes. The choice of the bisection direction has a critical influence on the efficiency of the procedure.

First, it is forbidden to bisect in any direction i for which

$$\Delta_i < 2\ tol_i^w \tag{5.171}$$

where tol_i^w $(i = 1, ..., s)$ is a minimal tolerance on the width in direction i, necessary to ensure the termination of the algorithm. If no error information is available, the direction associated with the largest normalized width Δ_i/tol_i^w is bisected. If the error analysis has been performed, the maximal value of $\widehat{e}_i$ defines the bisection direction.

To conclude, the adaptive decomposition of the configuration space limits the difficulties associated with the curse of dimensionality, and it is especially valuable for complex mechanisms with high-dimensional configuration space.

5.6.4 Additional parameters of the model

The vector $\mathbf{f}$ can collect the components of $\overline{\mathbf{M}}$, $\overline{\mathbf{h}}$, $\overline{\mathbf{K}}$, and $\boldsymbol{\Psi}^{g\boldsymbol{\theta}}$, necessary to describe the dynamics associated with the modal coordinates. However, any other relevant information can also be included in this vector, for instance, the modal contributions of the gravity forces, the amplitudes of the mode shapes at a sensor coordinate, etc.

5.7 Summary of the reduction procedure

After the detailed description of the reduction algorithm, and before considering a couple of applications, let us present an overview of the concepts proposed in this chapter, which are illustrated in Figure 5.16.

At an undeformed configuration with zero velocities, the linearized equations are formulated for the assembled mechanism, and exploited for the local mode synthesis. The local modes are consistent with a nonlinear Global Modal Parameterization (GMP), in the sense that they are connected with its Jacobian.

From an analysis of the potential and kinetic energies, the Lagrange equations are formulated, and the components of the stiffness matrix, of the mass matrix, and of the gyroscopic tensor are identified using the local reduction algorithm. The approximation technique allows a nonlinear, simplified and portable expression of the reduced model.

5.8 Applications

Two applications are treated in this chapter: an academic flexible four-bar mechanism, and a rigid parallel-kinematic machine-tool, called Orthoglide.

5.8.1 Four-bar mechanism

Figure 5.17 illustrates a four-bar mechanism with large configuration changes. The mechanism is controlled by a motor at the lower left hinge, so that the rigid motion is parameterized with the hinge angle θ. Configurations (a) and (e) are associated with actuator singularities, and they define the limits of our study in the configuration space (actually, the configuration space of interest is restricted to $\theta \in [-1.75, 1.75]$ rad). At the singular configurations, the actuator is not able to control the motion of the mechanism, and the aligned links may bend upward or downward, depending on other external forces. It is noticeable that another actuator location would lead to another actuator singularity.

We consider that the operations of a tool cause vertical loads on the upper right hinge, and one constraint mode is therefore associated with the vertical displacement z_e of that point. Three internal modes are also selected to represent the deformations of the mechanism. The initial Finite Element model contains 91 dofs, whereas the reduced model involves 5 modal coordinates (1 rigid mode, 1 constraint mode, 3 internal modes), which are represented for configuration (d) in Figure 5.18. Two internal modes have an out-of-plane deformation.

For this one-dimensional configuration space, the Lagrange interpolation technique is selected and different versions of the reduction algorithm are tested.

First, the configuration space is decomposed into 32 boxes according to a regular grid, and the mode tracking algorithm is disabled. The natural frequencies of the selected internal modes are plotted in Figure 5.19. On the left plot, non-

GLOBAL LEVEL

LOCAL LEVEL
undeformed configuration, zero velocity

$$\mathbf{M}(\mathbf{q})\,\ddot{\mathbf{q}} + \mathbf{\Phi}_\mathbf{q}^T\,\boldsymbol{\lambda} - \mathbf{g}(\mathbf{q},\dot{\mathbf{q}},t) = \mathbf{0}$$

$$\mathbf{\Phi}(\mathbf{q}) = \mathbf{0}$$

$$\rightarrow \quad \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \triangle\ddot{\mathbf{q}} \\ \triangle\ddot{\boldsymbol{\lambda}} \end{bmatrix} + \begin{bmatrix} \mathbf{K}_t & k\,\mathbf{\Phi}_\mathbf{q}^T \\ k\,\mathbf{\Phi}_\mathbf{q} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \triangle\mathbf{q} \\ \triangle\boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_{ext} \\ \mathbf{0} \end{bmatrix}$$

$$\downarrow$$

$$\overline{\mathbf{u}} = \boldsymbol{\rho}(\boldsymbol{\theta}) + \mathbf{\Psi}^{\mathbf{u}\delta}(\boldsymbol{\theta})\,\boldsymbol{\eta}^\delta \qquad \leftrightarrow \qquad \begin{aligned} \triangle\overline{\mathbf{u}} &= \begin{bmatrix} \mathbf{\Psi}^{\mathbf{u}\boldsymbol{\theta}} & \mathbf{\Psi}^{\mathbf{u}\delta} \end{bmatrix} \begin{bmatrix} \triangle\boldsymbol{\theta} \\ \boldsymbol{\eta}^\delta \end{bmatrix} \\ &= \mathbf{\Psi}^{\mathbf{u}\boldsymbol{\eta}}\,\triangle\boldsymbol{\eta} \end{aligned}$$

$$\downarrow \qquad\qquad \downarrow$$

$$\mathcal{V} = \tfrac{1}{2}\,\boldsymbol{\eta}^{\delta T}\,\overline{\mathbf{K}}^{\delta\delta}(\boldsymbol{\theta})\,\boldsymbol{\eta}^\delta$$

$$\mathcal{K} = \tfrac{1}{2}\,\dot{\boldsymbol{\eta}}^T\,\overline{\mathbf{M}}(\boldsymbol{\eta})\,\dot{\boldsymbol{\eta}}$$

$$\rightarrow \quad \begin{aligned} \overline{\mathbf{M}} &= \mathbf{\Psi}^{\mathbf{q}\boldsymbol{\eta}T}\,\mathbf{M}\,\mathbf{\Psi}^{\mathbf{q}\boldsymbol{\eta}} \\ \overline{\mathbf{K}}^{\delta\delta} &= \mathbf{\Psi}^{\mathbf{q}\delta T}\,\mathbf{K}_t\,\mathbf{\Psi}^{\mathbf{q}\delta} \\ \overline{\mathbf{h}} &= \overline{\mathbf{h}}^{\boldsymbol{\chi}} + \overline{\mathbf{h}}^{gyr} \end{aligned}$$

$$\downarrow \qquad\qquad \swarrow$$

Approximation

$$\overline{\mathbf{M}}\,\ddot{\boldsymbol{\eta}} + (\overline{\mathbf{h}}.\dot{\boldsymbol{\eta}})\,\dot{\boldsymbol{\eta}} + \begin{bmatrix} \mathbf{0} \\ \overline{\mathbf{K}}^{\delta\delta}\,\boldsymbol{\eta}^\delta \end{bmatrix} = \mathbf{\Psi}^{\mathbf{q}\boldsymbol{\eta}T}\,\mathbf{g}_{ext}$$

Figure 5.16: Overview of the model reduction method.

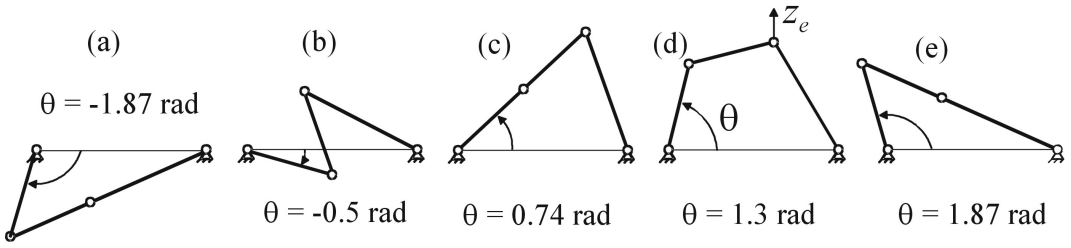


Figure 5.17: Large configuration changes of the four-bar mechanism

smooth variations are observed around $\theta = 0.7$ rad. They can be attributed to the presence of additional low-order modes which interfere with the selected internal modes. The algorithm selects the three modes with the lowest frequencies, so that discontinuities are observed whenever an eigenvalue crossing occurs.

This situation is of course not acceptable, since it leads to a non-consistent parameterization of the motion. The mode tracking strategy is able to remedy this situation, as attested by the results on the right plot in Figure 5.19. The selected modes are tracked in the configuration space, despite the presence of other low-frequency modes.

For the rigid mode, Figure 5.20 illustrates the variations of the equivalent mass and gyroscopic tensor in the configuration space. A vertical asymptote is expected close to the extreme singular configurations. Mathematically, the components of the rigid modes grow to infinity at those points, which explains the phenomenon. From the actuator point of view, the mechanical blocking at the singularity is equivalent to an infinite inertia. The relative error resulting from the approximation strategy is also presented:

$$err_i(\theta) = \frac{\left\| f_i(\theta) - \widehat{f}_i(\theta) \right\|}{\| f_i(\theta) \|} \tag{5.172}$$

A remarkable accuracy is obtained away from the singularities: the error is around 0.1% for the mass, and around 0.5% for the gyroscopic tensor. In particular, the errors vanish at every grid points. However, at the singularity, low-order polynomials defined on a regular grid are not able to represent the stiff behavior of the system. It is also observed that the gyroscopic tensor is connected with the gradient of the equivalent mass in the configuration space, in agreement with well-known principles of mechanics.

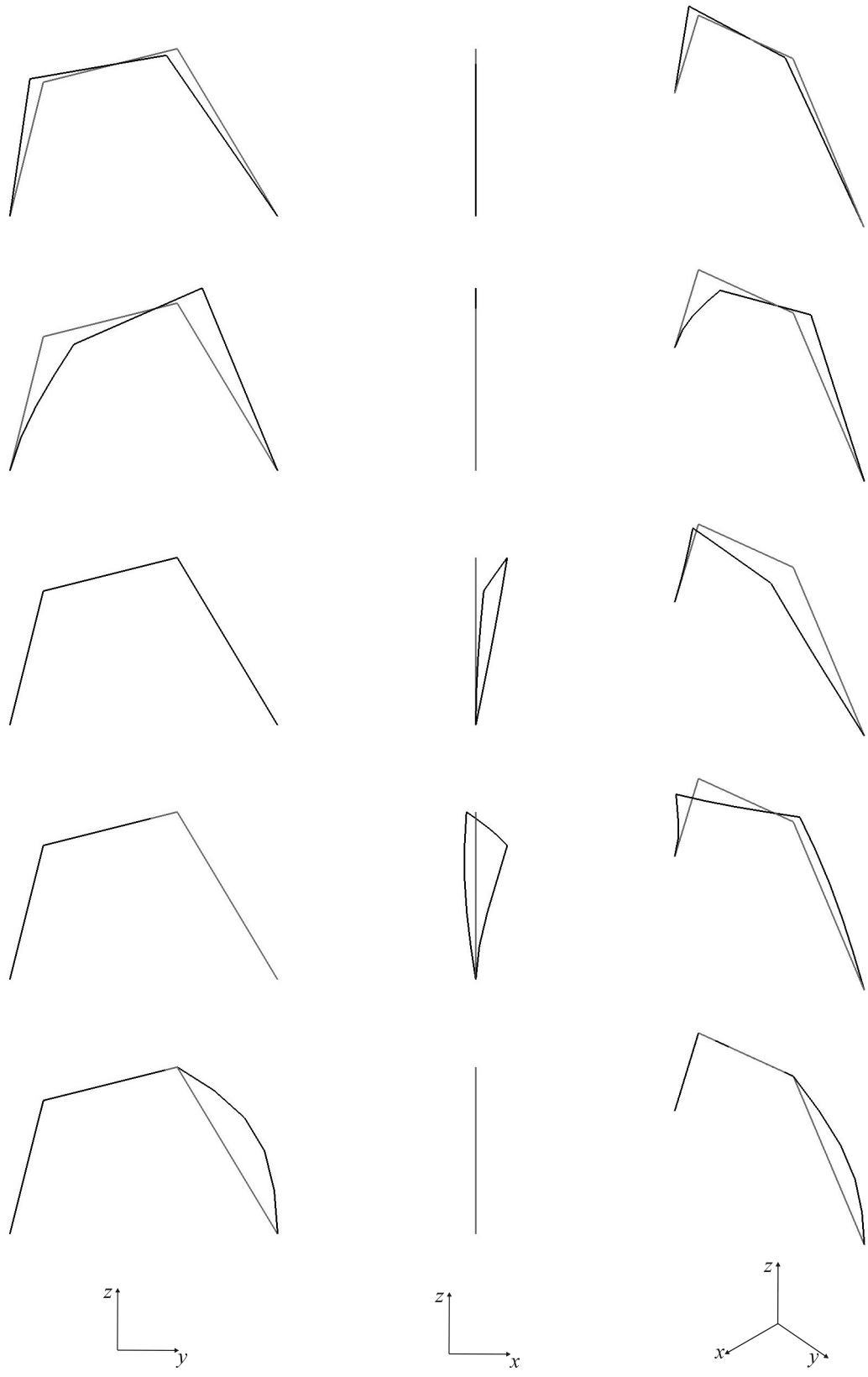


Figure 5.18: Mode shapes for configuration (d). From top to bottom: rigid mode, constraint mode (θ is fixed), and 3 internal modes (θ and z_e are fixed).

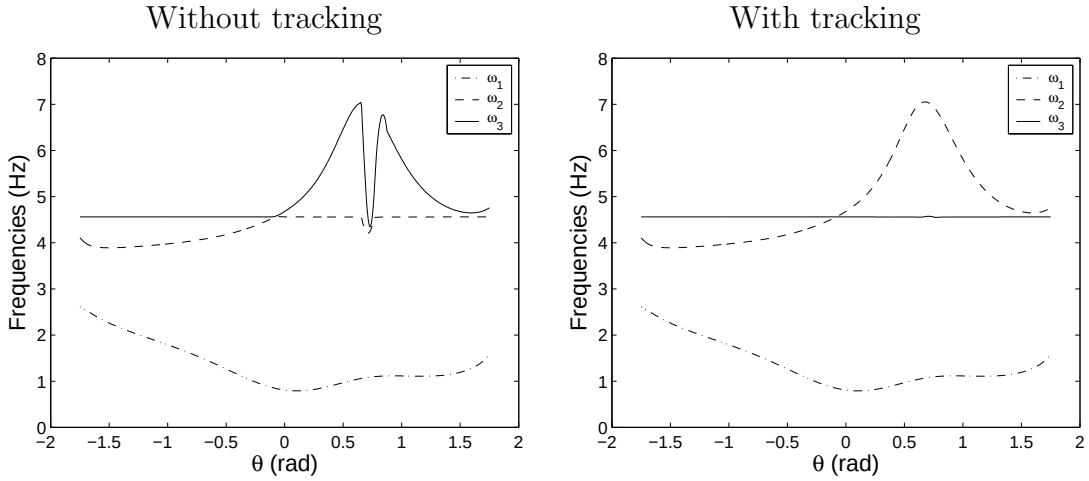


Figure 5.19: Natural frequencies in the configuration space - importance of the tracking strategy (obtained with a regular grid discretization).

The equivalent stiffness associated with the constraint mode is analyzed in Figure 5.21. The stiffness takes very large values around configuration (c) represented in Figure 5.17. In this configuration, the two left links are aligned, and any vertical motion z_e induces traction or compression efforts, involving tremendous strain energy. As a result of the large stiffness of the constraint mode, the vertical deflection of the effector is almost blocked.

In order to achieve a better trade-off between the number of boxes and the accuracy of the approximation, the adaptive strategy for the grid definition has been applied. The tolerance for the relative error is set to $tol^e = 0.003$ and the minimal box width $tol_1^w = 0.03\ rad$. The comparison between the regular and the adaptive grid is presented in Figure 5.22. The adaptive grid consists in 37 boxes (only 5 more than the regular grid), and it is refined close to the singular configurations, and close to configuration (c). The benefits of this strategy clearly appears in Figure 5.23, where inertia characteristics and relative errors are compared. For the adaptive strategy, the accuracy is strongly increased near the singularities, and small errors are tolerated in other parts of the configuration space, allowing a coarser discretization. Figure 5.24 reveals the ability of this approach to deal with the non-smooth behavior of the stiffness of the constraint mode around configuration (c). The regular grid leads to a bad approximation in a large domain around the peak, which may strongly affect the quality of the

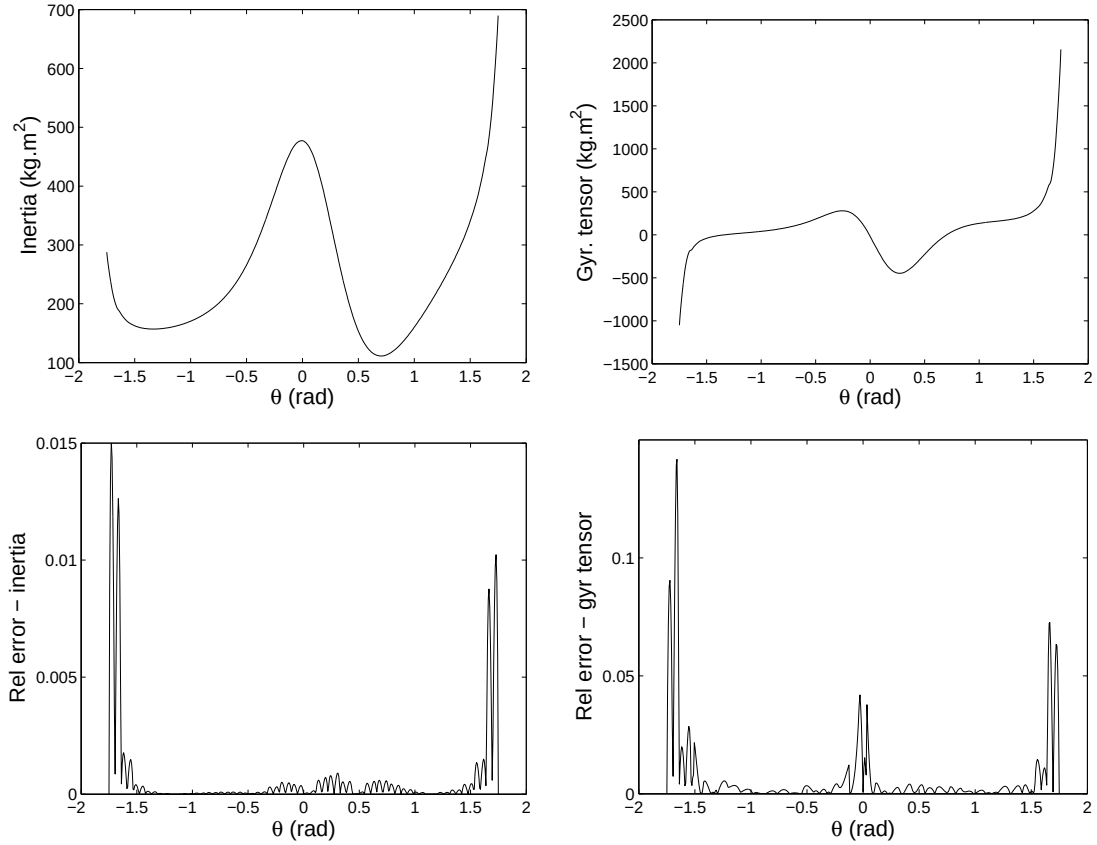


Figure 5.20: Regular grid with tracking strategy: variations of the equivalent inertia associated with the rigid mode $\overline{\mathbf{M}}_{\boldsymbol{\theta\theta}}(1,1)$, and of the gyroscopic tensor $\mathbf{h}(1,1,1)$.

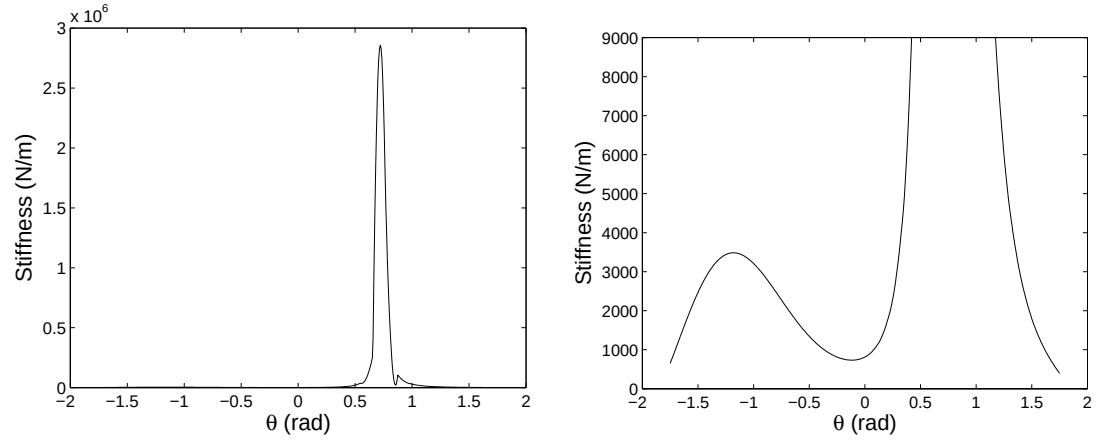


Figure 5.21: Regular grid with tracking strategy: equivalent stiffness associated with the constraint mode. The picture on the right is a zoom.

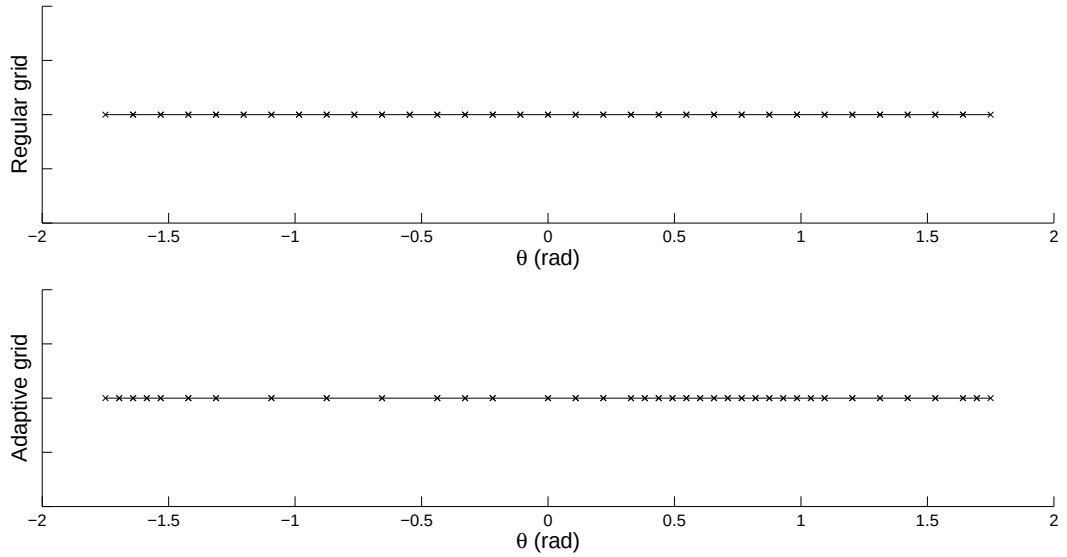


Figure 5.22: Regular and adaptive grid.

model.

Additional information can be recorded in the reduced model. For instance, the modal contribution of the gravity forces $\mathbf{g}^{grav}$ and the amplitude of the rigid modes for the constraint dof ($\boldsymbol{\Psi}^{g\boldsymbol{\theta}}$) are plotted in Figure 5.25.

From this example, we conclude that the reduced-order modeling technique is able to capture accurately the nonlinear changes in the stiffness and inertia properties. The mode-tracking strategy is essential to guarantee the consistency of the results. Close to the singular configurations, the adaptive configuration space decomposition is valuable to reduce the approximation error and to optimize the computational resources. In order to demonstrate the generality of the approach, a mechanical system with a 3-dimensional configuration space is considered in the next example.

5.8.2 The Orthoglide

The Orthoglide (Figure 5.26) is a parallel kinematic machine-tool designed at the IRCCyN research center (Nantes, France) for high-speed machining [WC00, CWA00, GKCW02, CWM02, CW03]. Three orthogonal linear actuators, fixed to the frame, control the 3-dimensional motion of the effector, and an isotropic behavior is obtained in the center of the workspace, where the 3 links are parallel

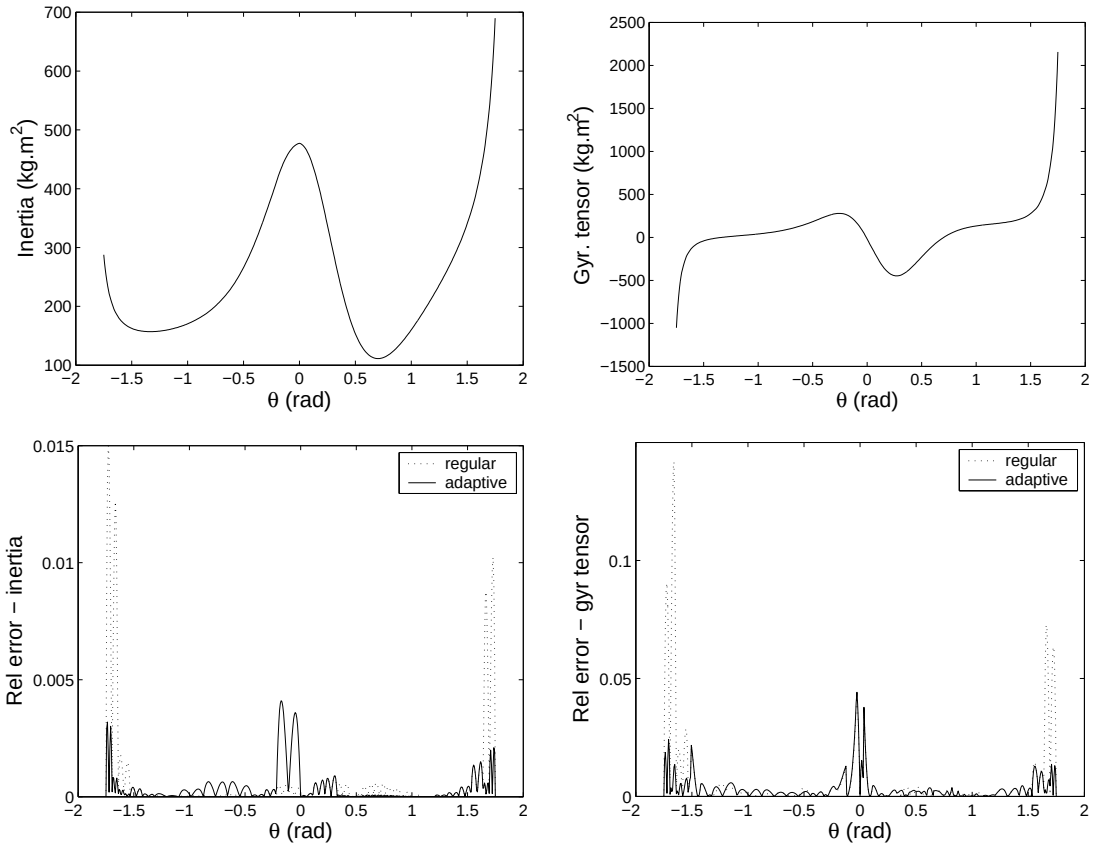


Figure 5.23: Comparison adaptive/regular grid: inertia components.

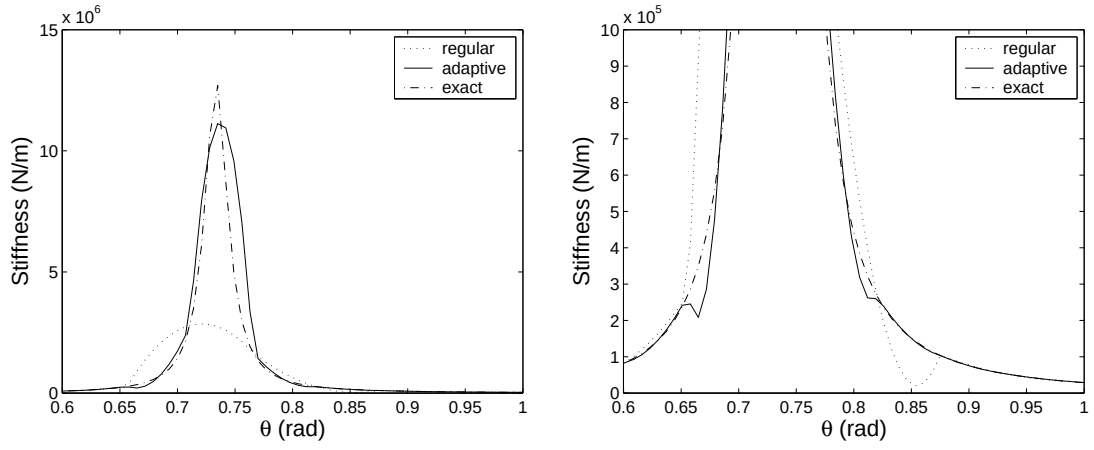


Figure 5.24: Comparison adaptive/regular grid: stiffness of the constraint mode for $\boldsymbol{\theta} \in [0.6\ 1]\ rad$.

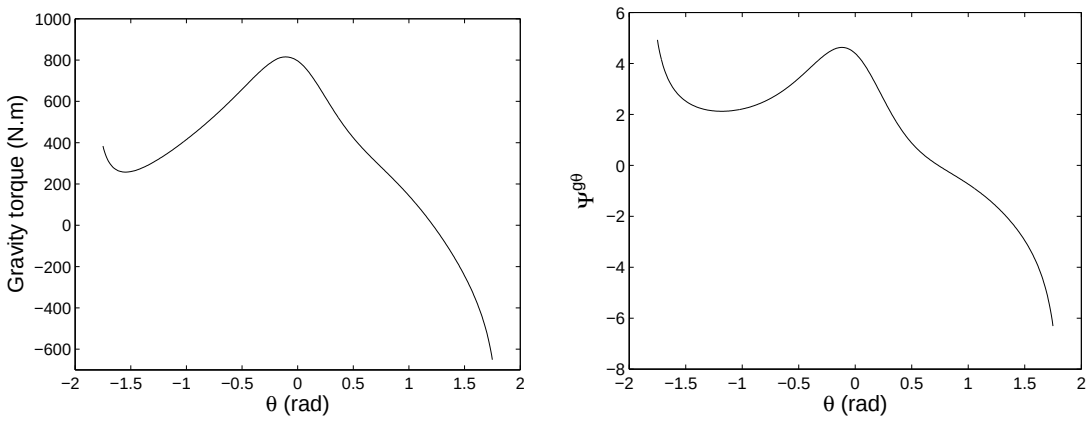


Figure 5.25: Additional information recorded in the model: equivalent gravity force exerted on the rigid dof, and amplitude of the rigid mode for the constraint dof $\boldsymbol{\Psi}^{g\boldsymbol{\theta}}(1,1)$.

to the inertial frame axes. The effector is connected to the actuators through a set of three parallelogram mechanisms, so that its orientation is kept constant.

A simplified model is constructed, assuming rigid links, and ideal joints. Since we are especially interested in the dynamic relation between the actuators and the translation of the effector, the parallelogram mechanism is modeled as a rigid link, connected to the actuator by a universal joint, and to the effector by a spherical joint, see Figure 5.27. Hence, the initial model involves 27 dofs: 15 generalized coordinates, and 12 Lagrange multipliers. In this case, the objective of the reduction is to formulate an equivalent model in term of the three independent actuator coordinates. This problem could be equivalently solved according to an extended constraint elimination method [BDG06]. The resulting inverse dynamic model is especially well-suited for the design of a computed-torque controller: the model predicts the required actuator forces and torques to produce a pre-specified motion.

Chablat *et al.* [CWM02] analyzed the workspace of the Orthoglide using interval analysis concepts. Here, the motion of each actuator is limited to the interval [-0.08, 0.26] m and the corresponding workspace does not contain any singular configuration; two extreme configurations are illustrated in Figure 5.28.

The reduced mass matrix and gyroscopic tensor are constructed. The approximation in the configuration space is done with quadratic and Lagrange poly-

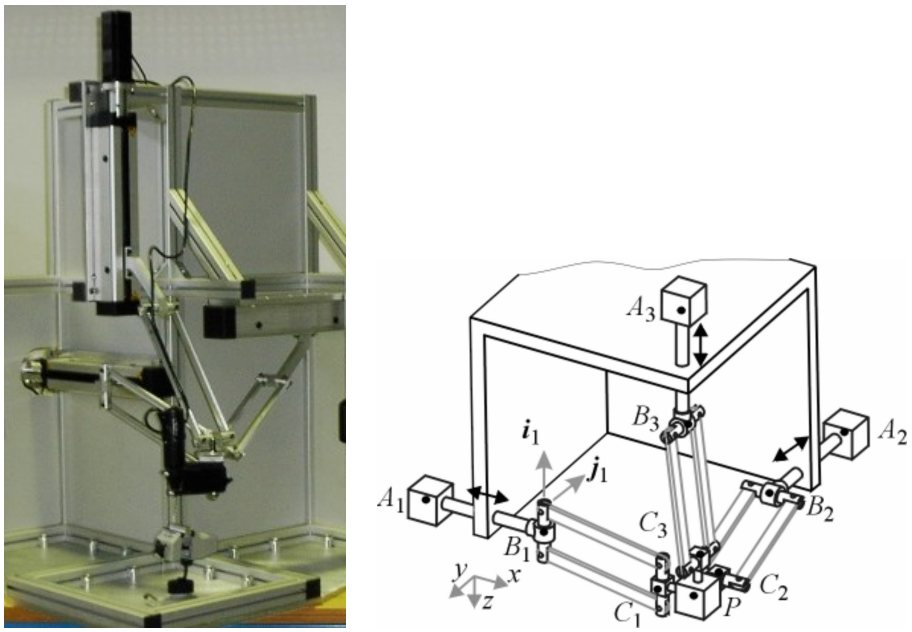


Figure 5.26: Prototype of the Orthoglide and conceptual kinematic representation. The length of the links is approximately 45 cm.

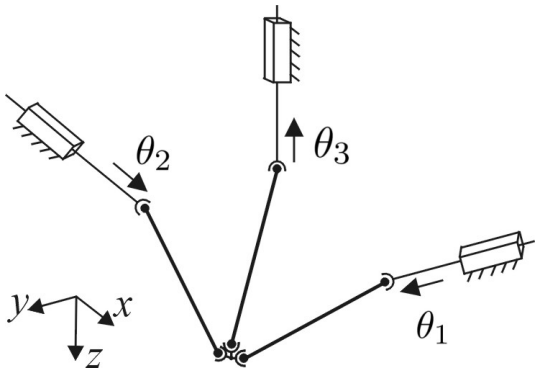
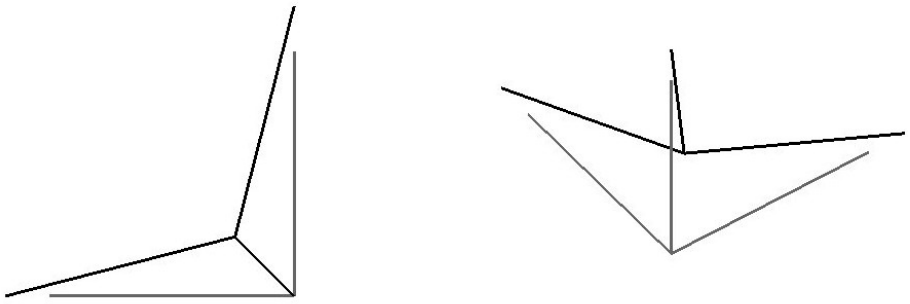


Figure 5.27: Model of the Orthoglide. The fixation of the links on the actuator slider is modeled as a universal joint to prevent their axial rotation.

$\boldsymbol{\theta} = [-0.08 \quad -0.08 \quad -0.08]^T \; m$:



$\boldsymbol{\theta} = [0.26 \quad 0.26 \quad 0.26]^T \; m$:

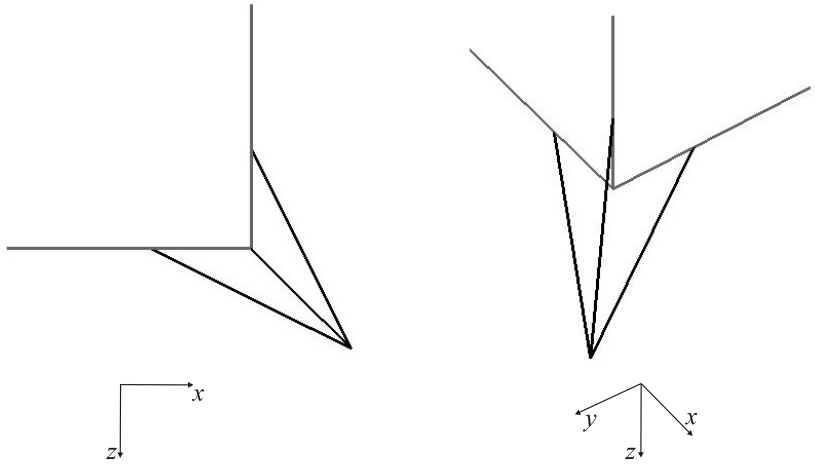


Figure 5.28: Extreme configurations of the Orthoglide. In grey lines, the configuration $\boldsymbol{\theta} = [0 \quad 0 \quad 0]^T \; m$.

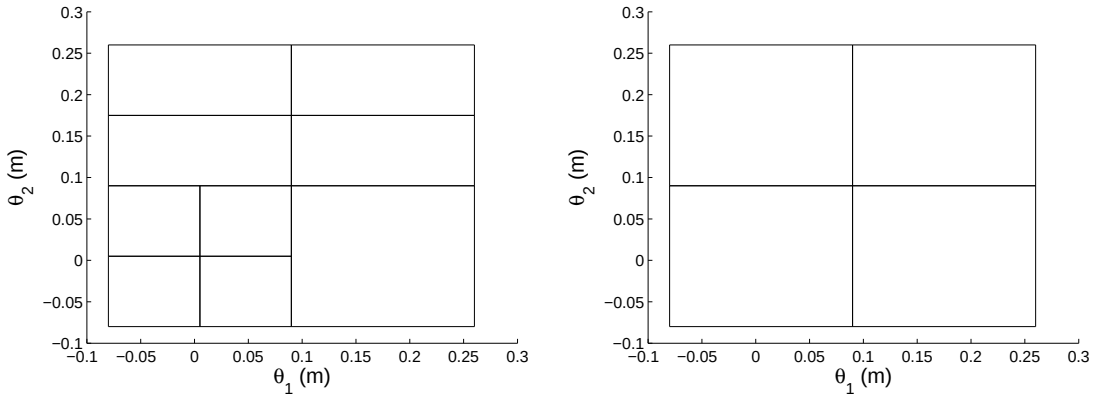


Figure 5.29: Configuration space discretization: Quadratic and Lagrange polynomials. Configuration space cut in the plane $\theta_3 = -0.08\ m$.

nomials. The tolerance on the relative error tol^e is fixed to 0.5%, the minimal box width tol_i^w is fixed to 0.05 m ($i = 1, 2, 3$). The discretization in the configuration is illustrated in Figure 5.29. Quadratic polynomials require 22 boxes, whereas 8 boxes are sufficient for Lagrange polynomials. Therefore, each component of the reduced model requires the storage of 220 floating-point numbers for quadratic polynomials, and of 216 numbers for Lagrange polynomials.

The variations of the inertia parameters along a straight line in the configuration space are presented in Figure 5.30. The quadratic polynomial is not appropriate, since significant discontinuities are observed. The results would be improved using smaller tolerances tol^e and tol_i^w, but it would lead to an exaggerated discretization of the configuration space. Using Lagrange polynomials, the relative error is kept below 0.5% for the mass, and below 3% for the gyroscopic tensor.

As a conclusion, the reduction method is applicable to a spatial mechanism with 3 kinematic dofs. The nonlinear variations of the inertia properties are significant but smooth, since the actuator singularities are avoided. Thus, piecewise polynomials lead to an efficient representation of the dynamic model.

5.9 Summary and concluding remarks

This chapter proposed a modeling tool for the compact and closed-form representation of complex flexible multibody systems with parallel topology. Following

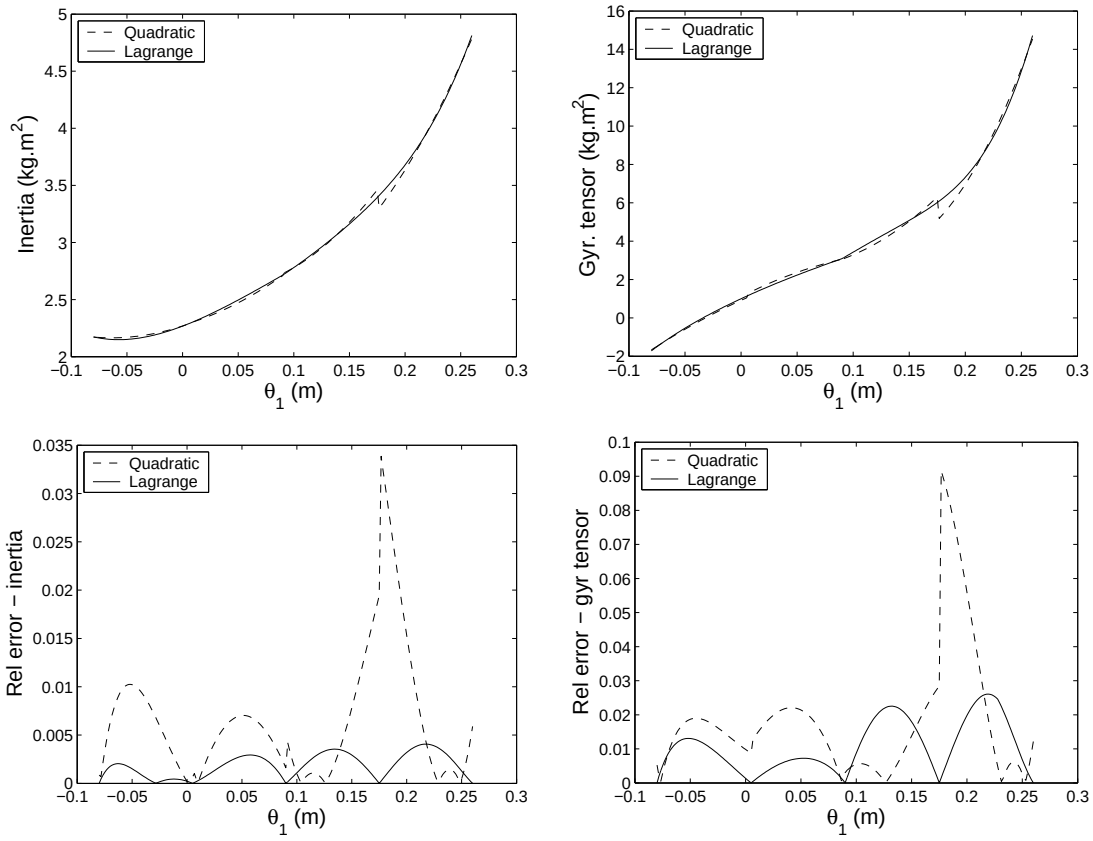


Figure 5.30: Variations of the inertia components from the extreme configuration $\boldsymbol{\theta} = [-0.08 \quad -0.08 \quad 0.26]^T \ m$ to $[0.26 \quad 0.26 \quad -0.08]^T \ m$. On the left, the equivalent mass associated with the first rigid mode $\overline{M}_{11}^{\boldsymbol{\theta\theta}}$, on the right the component $\overline{h}_{111}$.

a model reduction approach, the method relies on the following assumptions:

- the system is conservative and holonomic, no pre-stressing effect is considered,
- the singularities are avoided in the configuration space,
- the elastic behavior is linear, the reduced stiffness only depends on the kinematic configuration $\boldsymbol{\theta}$, and centrifugal stiffening effects are neglected,
- the reduced parameterization is defined in an optimal way on the basis of the linearized equations around any kinematic configuration with zero velocities, *i.e.*, it is optimized for quasi-static configuration changes.

The resulting model is described concisely according to a Global Modal Parameterization, which is free from kinematic constraints. The clear physical interpretation of the modal coordinates guarantees the consistency of the formulation. The reduced model is represented by a piecewise polynomial function, with the advantages of computational efficiency and portability. In the reduction procedure, the accuracy loss is localized at three levels:

- the truncation of the modal basis (this source of error disappears when considering a rigid mechanism),
- the approximation in the configuration space,
- the elimination of nonlinear phenomena associated with large deformations.

The procedure is systematic, and the user only needs to provide high-level information:

- a standard Finite Element model of the mechanism,
- the partitioning into rigid, constraint, and internal dofs,
- the number of internal modes,
- an inner subpaving approximation of the configuration space,
- the tolerance on the approximation error, and the minimum size of the boxes.

The mode shapes are deduced automatically, even if the mechanical topology is complex and if the distribution of physical properties in the flexible bodies is non-uniform. This is a definite advantage over assumed-mode techniques, where the appropriate definition of the modes and boundary conditions is often intricate.

In this chapter, two examples have been successfully analyzed: a flexible four-bar mechanism, and a rigid parallel kinematic machine-tool. In chapter 6, the reduced-order model of a flexible manipulator will be exploited for the design of a control algorithm.

The dofs appearing in the final model are directly associated with the actuators, the sensors, and other components of the mechatronic system. The method can be exploited for control design in (at least) three different ways:

- online implementation of a feedback controller: computed torque control of a rigid mechanism, active vibration control (see chapter 6),

- offline numerical optimization of a control law: trajectory generation, inverse dynamics, training of a feedback action,

- construction of a more structured model, in order to fit well-established control theories: polytopic linear model, model based on linear fractional transformation, etc. Then, the reduction procedure makes the database $\boldsymbol{\theta}^{(i)} - \mathbf{f}^{(i)}$, $i = 1, ..., r$ available for the external approximation algorithm, and the configuration space decomposition can be interpreted as an adaptive inspection algorithm, which optimizes the information content of the database.

A priori knowledge for improved configuration space inspection

One may object that the size of the database and the computational effort to build the reduced-order model increase exponentially with respect to the dimension of the configuration space. At the end of this chapter, two direct extensions of the original method are discussed to overcome this problem.

The reduction method has been presented assuming that the model should be available in the whole configuration space. But, in many control applications, the trajectory (or the desired trajectory) of the mechanism is known in advance. The inspection strategy can be restricted to the configurations along the trajectory,

which is parameterized using the arclength, the time, or a pseudo-time variable. The approximation then becomes a one-dimensional problem, which can be solved with a high efficiency. The resulting model is a linear time-varying model about the reference trajectory, if the gyroscopic effects are neglected.

The approximation strategy can benefit from an analysis of the configuration space symmetries. Indeed, some properties of the dynamic model may be invariant with respect to a group of transformations, so that the inspection can be restricted to a lower-dimensional part of the configuration space. For instance, if a mechanism is mounted on a straight slider, the whole dynamic model is certainly not affected by a translation on the slider. Our algorithm would behave nicely in such circumstances, since the interpolating polynomials are able to represent exactly the constant behavior along the translation coordinate, and no bisection shall occur in this direction (provided slight customization). Likewise, a mechanism mounted on a moving basis, *e.g.* a free-flying manipulator, has six rigid dofs associated with the motion of the basis, but the dynamic model represented in local axes is not affected by the overall motion. In this case, if an updated Lagrangian point of view is adopted for the orientation of the basis *in the reduced model*, the construction of the model can be restricted to the relative configuration changes with respect to the basis.

VI

Modeling, Simulation and Control of an Experimental Manipulator

In this chapter, the theoretical concepts presented previously are applied to the dynamic analysis and the control design of a lightweight manipulator called RALF, an acronym for Robot Arm Large and Flexible.

Different models are elaborated and utilized for frequency domain analysis, simulation and control design, as detailed in Table 6.1. Those models are defined either for the *mechanism*, for the *actuated mechanism* or for the whole *controlled mechanism*. An experimental validation is also conducted.

Guided by those models, a control strategy is developed and implemented for the flexible manipulator. According to a two-time-scale approach, a traditional joint-tracking controller is complemented with a fast indirect vibration controller. The vibration controller exploits the inertia forces of the manipulator in order to damp the flexible motion. Hence, it is an extension of inertial damping schemes developed for the control of macro/micro-manipulators [Geo02].

After a description of the test-bed, the modeling of the actuated mechanism is addressed in section 6.2, and an experimental validation is realized in the frequency domain. In section 6.3, the composite two-time-scale controller is designed. Section 6.4 deals with the performance analysis of the controlled mechanism: the predictions of various models are compared with experimental results.

	Mechanism	Actuated mechanism	Controlled mechanism
Full-order nonlinear			Time simulation
Reduced-order nonlinear	Control design		Time simulation
Reduced-order linear		Frequency response	Root locus Frequency response

Table 6.1: Different models and their use for dynamic analysis and control design. The lines of the tabular are associated with the type of mechanical model, the columns with the additional components of the mechatronic system.

6.1 Description of the test-bed

The manipulator Ralf, shown in Figure 6.1, is a long-reach manipulator operating in a vertical plane, that has been developed at the Georgia Institute of Technology. It was designed by Wilson [Wil86], and completed by Huggins [Hug88]. Ralf has two kinematic dofs, and it is hydraulically actuated.

The structure consists of two main links and a parallel actuation mechanism. The two main links are 3.05 m long and constructed from aluminium pipes, whereas the lighter actuation link is constructed from a rectangular aluminium tube. Thick sleeves of aluminium tubing connect the links to each other, to the actuators, and to the base. The joints are constructed from bronze bushings and steel shafts. The assembled manipulator structure without actuators and base weights approximately 45 kg, with its payload capacity of 27 kg. Hence, Ralf has a high payload to weight ratio, and it is stiff enough to achieve real-world applications. However, flexible effects in the aluminimum tubes affect the positioning accuracy.

Two hydraulic cylinders are used for the actuation of Ralf. The first moves the first link relative to the base, the second moves the second link relative to the first through the parallel mechanism. Each cylinder is controlled by a hydraulic servovalve. The oil is supplied at 1500 psi. The maximum cylinder velocities are 0.156 m/s for extension and 0.208 m/s for retraction.

Two linear position sensing transducers, fixed to the hydraulic cylinder, mea-

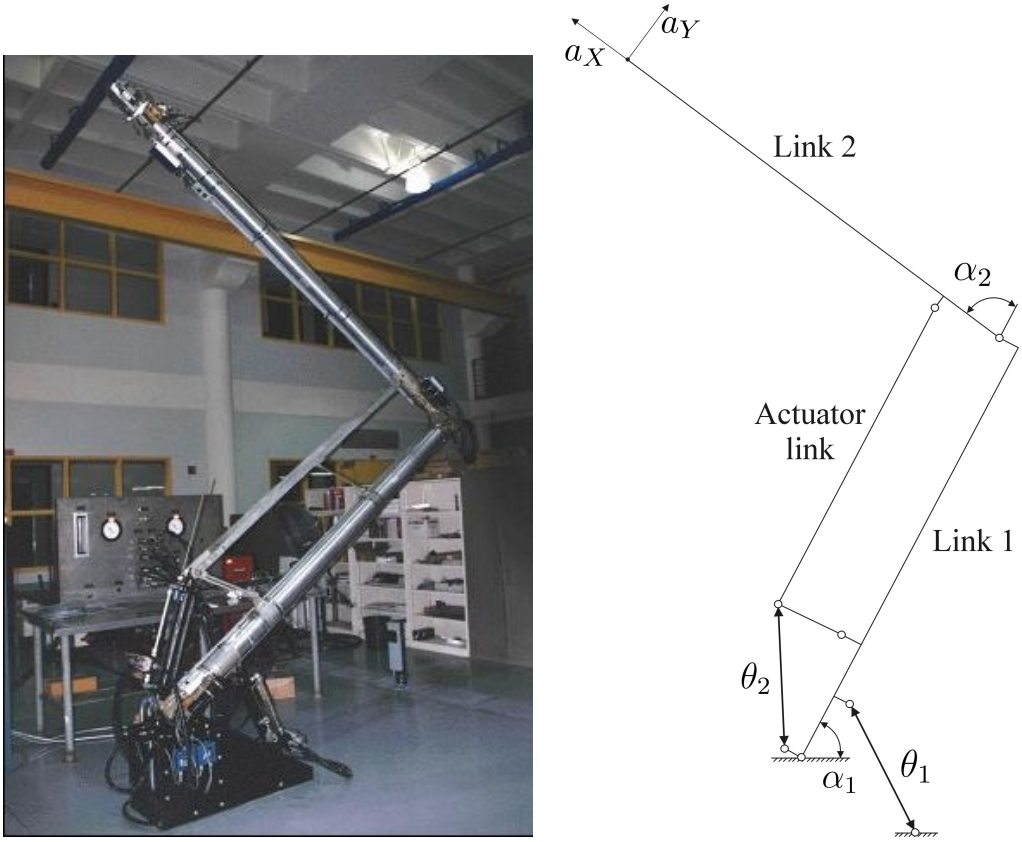


Figure 6.1: Ralf, coordinates of the actuators and the sensors.

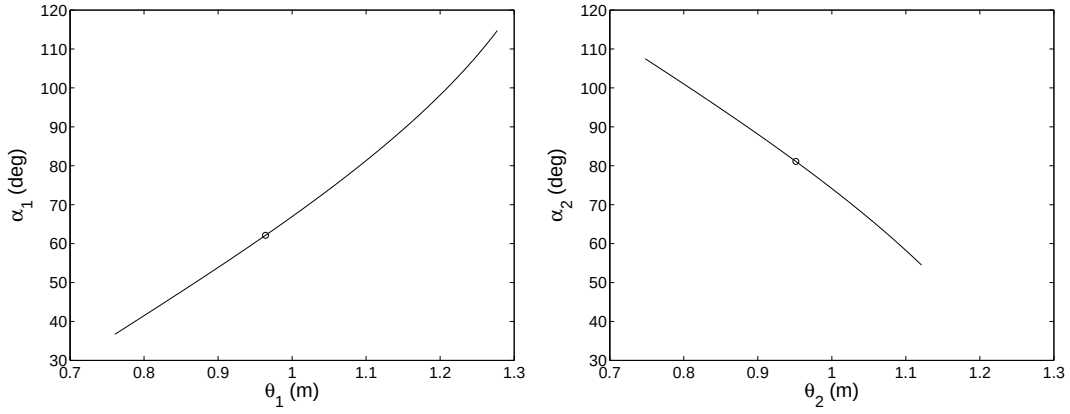


Figure 6.2: Relation between actuator coordinates θ_i and link angles α_i, $i = 1, 2$. The circle represents the home configuration.

sure the cylinder extension. Moreover, in order to detect the vibrations of the mechanism, two accelerometers have been placed at the tip, in two orthogonal directions.

The control law is implemented on a real-time computer, equipped with a National Instruments A/D board. The control law is developed in the Simulink environment, built using Real-Time workshop, and loaded on the real-time computer thanks to the XPC Target architecture.

Amplifiers and power supplies are in charge of generating the electrical signals for the hydraulic valves, as well as to amplify the accelerometer signals.

6.1.1 Workspace description

As seen in Figure 6.1, the kinematic configuration of Ralf can be described either using the relative angles α_1 and α_2 of the main links, or the actuator extensions θ_1 and θ_2. The relation between those coordinates is illustrated in Figure 6.2. The tip-accelerometers measure the absolute accelerations a_X and a_Y in body axes.

The allowed configuration space is defined by a rectangular box represented in Figure 6.3. In the same figure, 6 configurations of interest are mentioned; configuration 0 is the so-called *home position*.

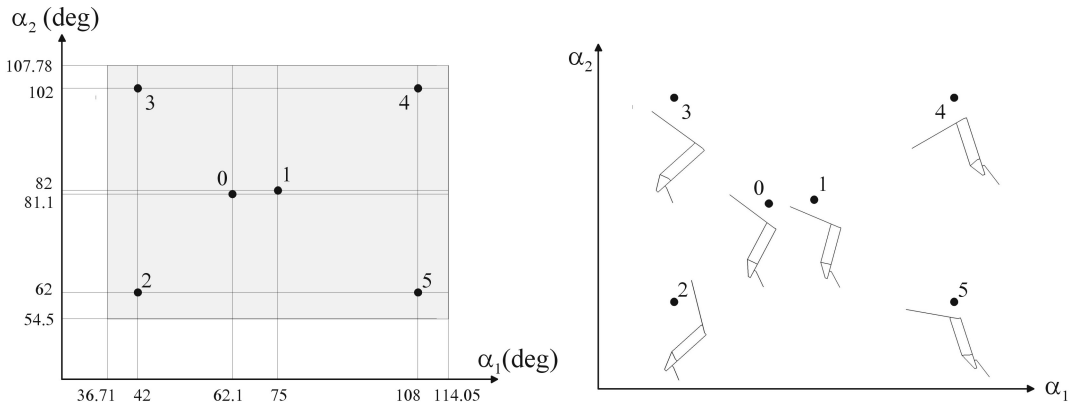


Figure 6.3: Configuration space of Ralf.

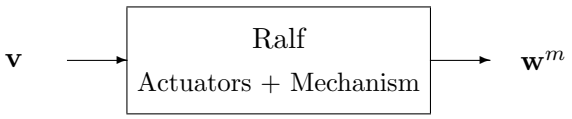


Figure 6.4: Ralf - input/output representation.

6.1.2 Transfer functions acquisition

In the experimental system, the inputs are the voltages of the actuators, and the outputs are the sensor measurements, *i.e.* the extensions of the hydraulic actuators and the tip accelerations. We define the voltage vector $\mathbf{v}$ and the measurement vector $\mathbf{w}^m$:

$$\mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}, \qquad \mathbf{w}^m = \begin{bmatrix} \theta_1^m \\ \theta_2^m \\ a_X^m \\ a_Y^m \end{bmatrix} = \begin{bmatrix} \boldsymbol{\theta}^m \\ \mathbf{a}^m \end{bmatrix} \tag{6.1}$$

An input/output view of the actuated mechanism is illustrated in Figure 6.4.

In order to establish the linearized transfer functions of the mechanism around different operating points, the actuators are independently controlled by low gain proportional controllers, and a swept-sine signal excites the system, as illustrated in Figure 6.5. The resulting information shall be used for the validation of the

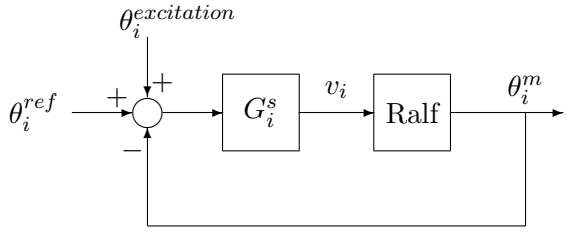


Figure 6.5: Independent SISO controller for each actuator ($i = 1, 2$). Low gain values $G_1 = G_2 = 100$.

mechanical model.

6.2 Model of the manipulator

6.2.1 Previous models of Ralf

Huggins [Hug88] and Lee [Lee90] achieved an experimental validation of an assumed-mode model based on Lagrangian dynamics. They considered a two-link model, with two vibration modes per beam. The definition of appropriate boundary conditions for each link turned out to be a difficult task. In order to verify the quality of the assumed-mode model, they developed and validated a linear Finite Element model. On this basis, improved shapes of the assumed-modes were obtained by component-mode synthesis. Despite small discrepancies observed with respect to experimental data, the assumed-mode model was sufficiently accurate for control design. The dynamic behavior of the actuators was investigated, it appeared to play a significant role.

Lee [Lee90] developed a Lagrangian dynamic model including the parallel drive mechanism, represented by nonlinear constraint equations. For simulation purpose, a constraint elimination technique based on a singular value decomposition was implemented. Through experimental work, Lee observed that the importance of the inertia forces quadratic in velocity is minor, due to the speed limitation of the hydraulic cylinder.

Magee [Mag91] achieved an experimental analysis of Ralf in the whole workspace. The structure was excited by the hydraulic actuators, and the tip motion was measured using accelerometers. The underlying assumed-mode model was based on

the Lagrangian method formulated by Book [Boo84]. In order to develop an input shaping strategy, a curve fitting method was applied to get the experimental values of the eigenfrequencies and damping ratio in the configuration space.

The design of Ralf evolved over the years, which explains substantial differences in the results obtained by those researchers. More recent investigations were realized by Obergfell [Obe98], who modeled Ralf as a serial two-link manipulator. Surprisingly low natural frequencies were observed, that could be hardly represented by the mechanical model. Obergfell concluded that the actuator dynamics should be responsible for those discrepancies. Introducing a more realistic model of the hydraulic actuators, significant improvements were obtained, even though the natural frequencies predicted by the model were still 20 % too high.

6.2.2 Structural model

Following the formalism presented in chapter 3, a nonlinear Finite Element model has been elaborated for Ralf. Compared with previous models, this approach allows to represent all the details of the mechanism, including the parallel actuation mechanism. The two main links and the actuator link are modeled using flexible beam elements, all other components are considered as rigid-bodies or lumped masses. For the details about the geometrical data and the computations of the different masses, we refer to the drawings in [Wil86] and to the numerous reports on Ralf, such as [Obe98]. The gravity forces are taken into account.

Various other components are present on Ralf, and their masses have been estimated and introduced in the model. The structural damping has been adjusted to a value around 1 % (actually, the dominant damping effect in the overall system comes from the actuators and the control system). The Finite Element model involves 164 dofs.

Around a fixed configuration, the dynamics of Ralf is reasonably represented by a linearized model, and a frequency-domain analysis is possible. The linearized model is reduced according to the standard component-mode technique: 2 rigid modes and 2 flexible modes are able to capture the essential dynamic behavior of the system in the frequency range 0-15 Hz. Higher order modes are above 30 Hz, and their participation to the dynamic response is neglected.

The experimental validation of this model relies on transfer functions from

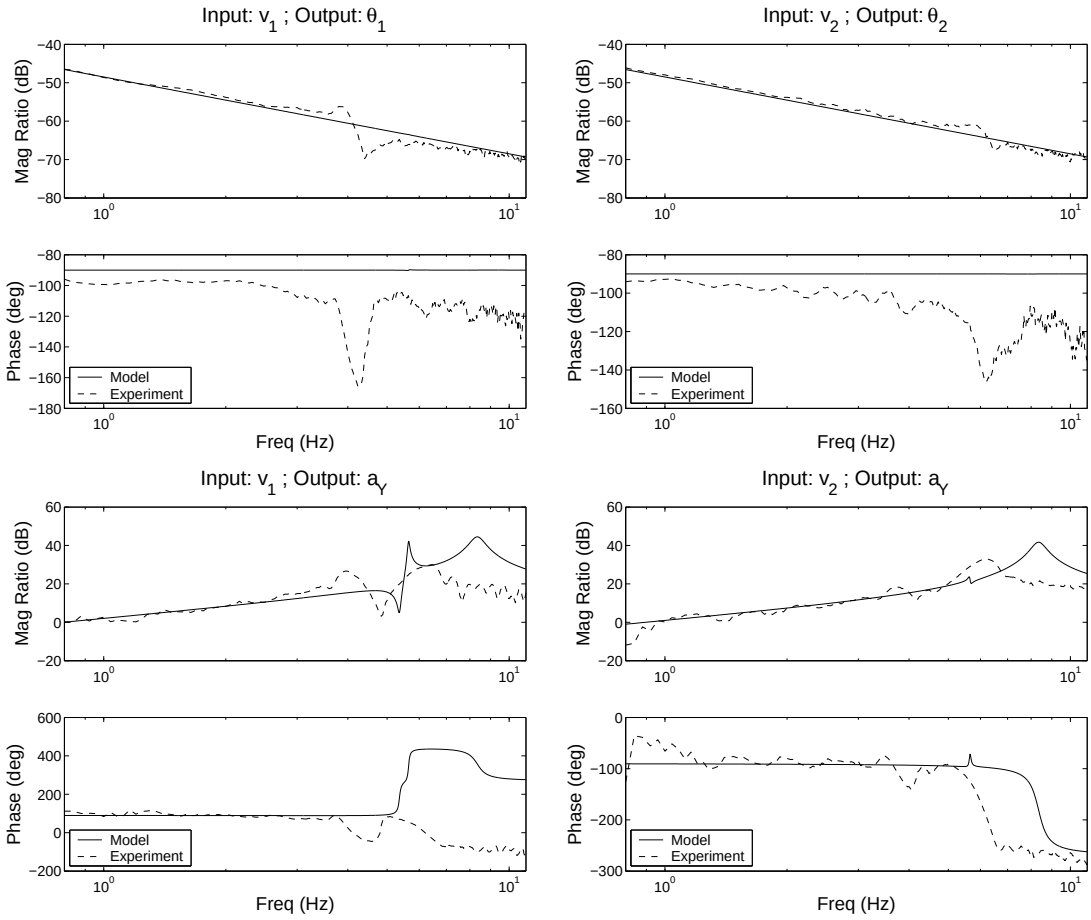


Figure 6.6: First actuator model (home configuration).

input voltages to linear sensors, and to accelerometers established for different configurations. A model of the hydraulic actuators is also needed at this level.

6.2.3 Validation and actuators models

First actuator model: velocity source

The experimental transfer functions from voltages to actuator displacements are plotted Figure 6.6. A velocity source model of the actuator is also considered, which behaves as an integrator:

$$\dot{\theta}_i = G_i^{hydr}\, v_i, \qquad i = 1, 2 \tag{6.2}$$

where θ_i is the displacement of the actuator i, v_i is the applied voltage, and G_i^{hydr} is the hydraulic gain, whose value has been fitted on the experimental data:

$$G_1^{hydr} = 0.0236, \qquad G_2^{hydr} = 0.0265 \tag{6.3}$$

This ideal model does not adequately capture the dynamics of the physical system in two ways:

- the actuator does not behave as a velocity source near the natural frequencies of the system,

- the predicted frequency responses from voltages to tip accelerations, also plotted in Figure 6.6, are not satisfactory.

For these reasons, and according to the idea of Obergfell [Obe98], a more elaborated model of the actuator is investigated, accounting for the limited dynamics of an intrinsic velocity feedback.

Second actuator model: intrinsic velocity feedback

In order to account for the intrinsic velocity feedback that regulates the flow through the valve of a hydraulic actuator, the following actuator model is proposed:

$$g_i^a = G_i^v \, (G_i^{hydr} \, v_i - \dot{\theta}_i), \qquad i = 1, 2 \tag{6.4}$$

where g_i^a is the force applied by the actuator, and G_i^v is a new constant to be determined. If $G_i^v \to \infty$, this model is equivalent to the velocity source model. For finite G_i^v, the transfer function between v_i and θ_i is affected by the dynamics of the mechanical system, especially near the resonances.

The value of G_i^v influences the transfer functions from voltages to actuator displacements, but unfortunately, the transfer function from voltages to tip acceleration is not improved, as shown in Figure 6.7, with the values:

$$G_1^v = 3.5e5, \qquad G_2^v = 3.0e5 \tag{6.5}$$

In particular, the predicted transfer functions are shifted to the high frequencies, which means that the model is probably too stiff. Therefore, let us consider the effect of a lumped flexibility at the level of the actuators.

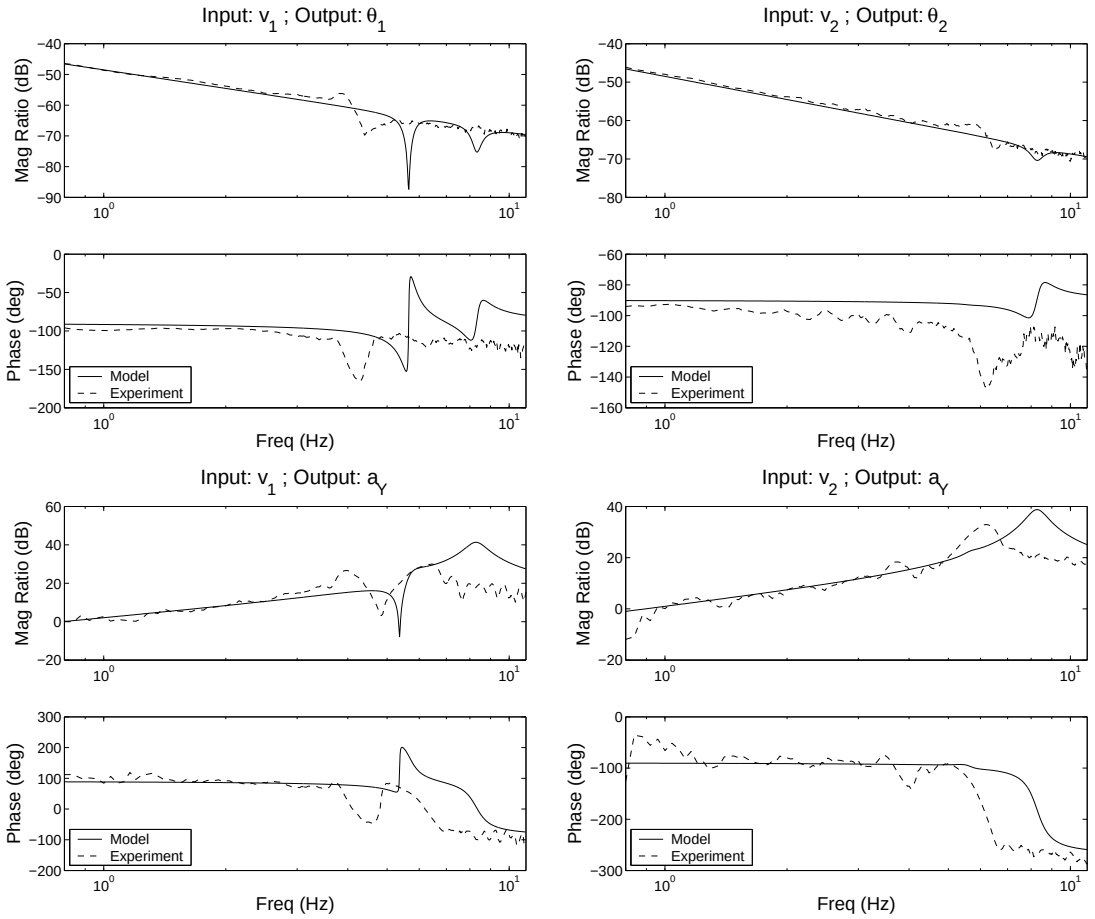


Figure 6.7: Second actuator model (home configuration).

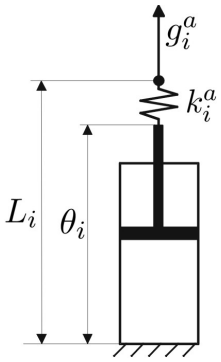


Figure 6.8: Actuator model including a lumped flexibility. g_i^a denotes the generated force, k_i^a is the finite stiffness of the actuator, θ_i and L_i the actuated dof and the total length.

Third actuator model: intrinsic velocity feedback and flexibility

In order to obtain a better matching between the model and the experiments, lumped flexible effects are considered in the actuators, see Figure 6.8. As in the previous section, the actuator forces are modeled according to equation (6.4), but the length L_i between the attachment points of the actuator is now different from the actuated dof θ_i.

It is not clear whether the measured displacement θ_i^m corresponds to L_i (non-collocated sensor/actuator configuration) or to θ_i (collocated sensor/actuator configuration). Hence, we follow a two-step identification procedure: (i) select the stiffnesses k_i^a in order to improve the transfer function v_i - **a**, (ii) compare the experimental transfer functions v_i - θ_i^m with the model transfer functions v_i - θ_i and v_i -L_i.

(i) The local stiffnesses are easily adjusted by the observation that actuator 1 has a dominant effect on the first mode, whereas actuator 2 essentially excites the second one. An iterative procedure results in the selection of the following stiffness values:

$$k_1^a = 3.5e6\ N/m^2, \qquad k_2^a = 3.0e6\ N/m^2 \tag{6.6}$$

For the set of configurations mentioned in Figure 6.3, the model and experimental transfer functions are plotted in Figure 6.9, 6.10 and 6.11. The correspondence is noteworthy, despite strong variations from one configuration to the other. In some configurations, the predicted frequency response are still slightly shifted to the high frequencies, but the model captures the dynamics with sufficient accuracy for control design.

(ii) The actuator transfer functions are plotted for the collocated and non-collocated cases in Figure 6.12. It seems that the model with collocated measurements is closer to the experimental results; it is selected for the following developments.

6.2.4 Nonlinear model reduction

The model of the actuated mechanism can be splitted into one model for the *actuator* and one model for the *mechanical system*, as illustrated in Figure 6.13, where the vector of actuator forces $\mathbf{g}^a = [g_1^a\ g_2^a]^T$ is computed according to formula (6.4). The actuator flexibility is included in the mechanical model.

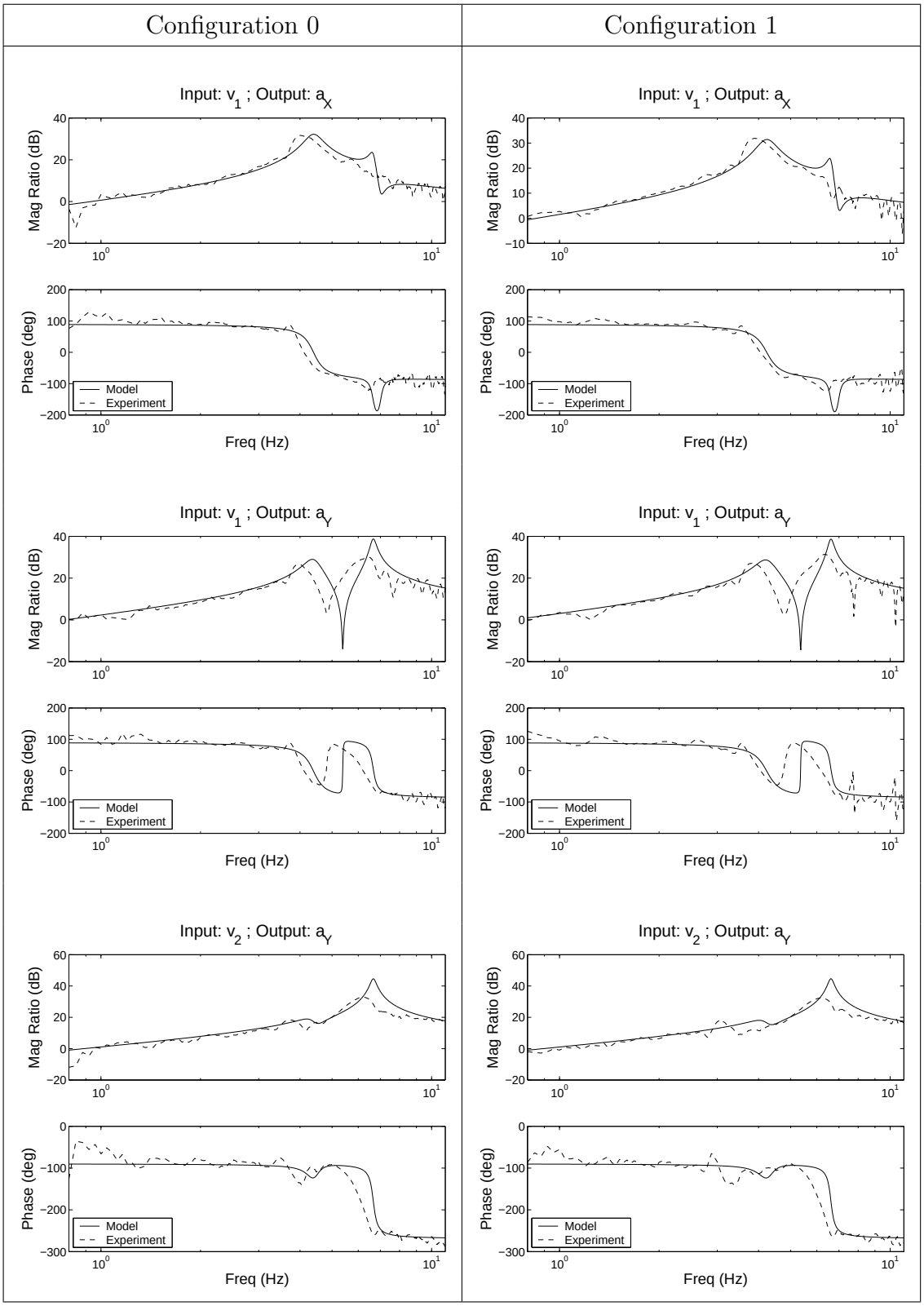


Figure 6.9: Experimental and model transfer functions, configurations 0 (home) and 1.

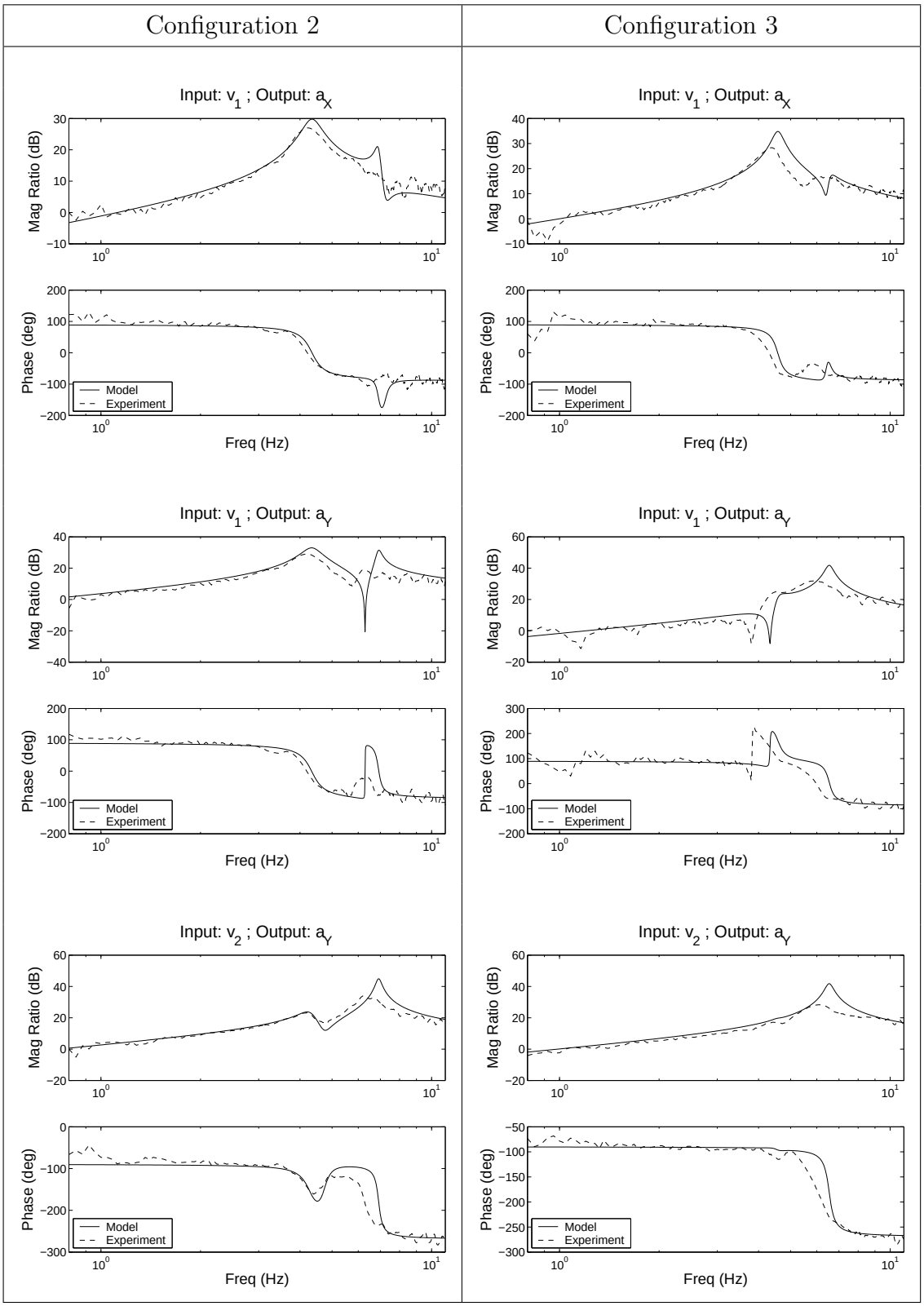


Figure 6.10: Experimental and model transfer functions, configurations 2 and 3.

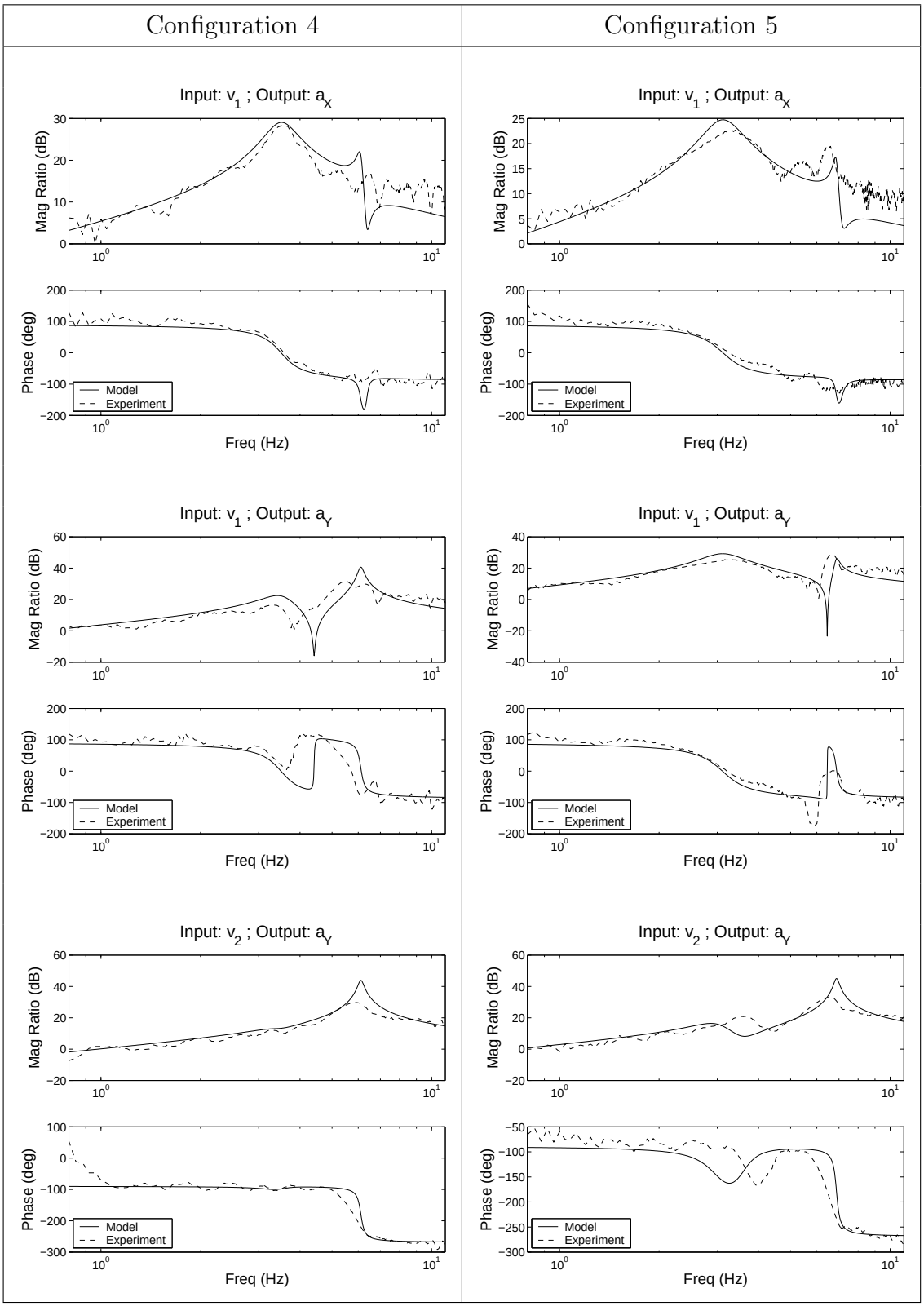


Figure 6.11: Experimental and model transfer functions, configurations 4 and 5.

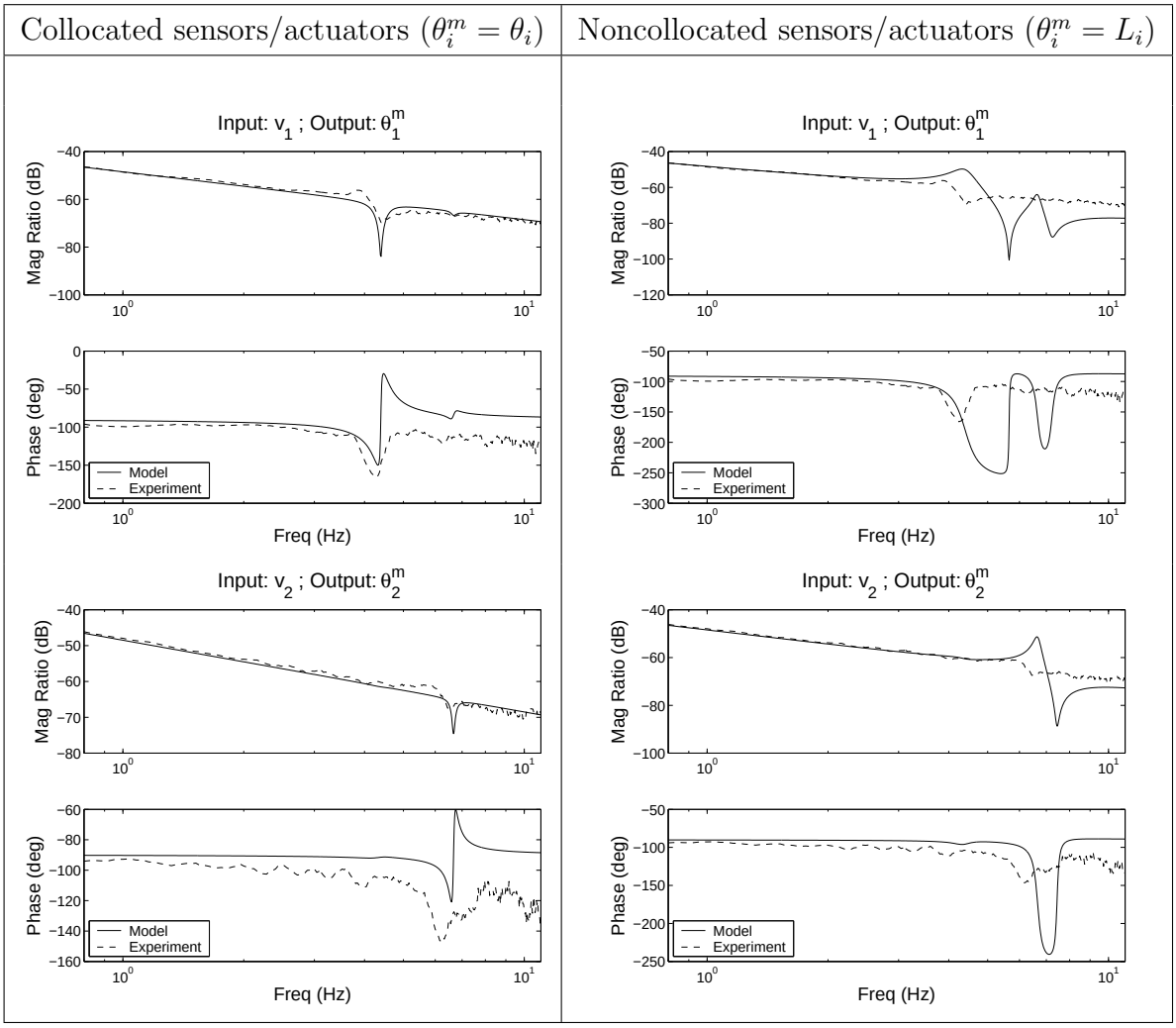


Figure 6.12: Third actuator model (home configuration).

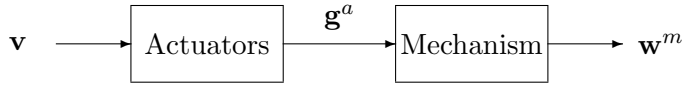


Figure 6.13: Mechanical and actuator model. The model of the mechanism includes the lumped stiffnesses at the level of the actuators.

In this section, a reduced *mechanical model*, valid in the whole configuration space is constructed according to the method presented in chapter 5. The two parameters of the rigid motion are naturally the actuated dofs $\boldsymbol{\theta} = [\theta_1 \ \theta_2]^T$. Two internal modes, with coordinates $\boldsymbol{\eta}^\delta = [\eta_1^\delta \ \eta_2^\delta]$ are sufficient to represent the dynamic behavior in the bandwidth of interest (0-15 Hz). The rigid and flexible mode shapes are illustrated in Figure 6.14.

The gyroscopic forces are neglected, which is a reasonable assumption according to Lee [Lee90]. Therefore, the reduced equations of motion are:

$$\begin{bmatrix} \mathbf{M}^{\boldsymbol{\theta\theta}} & \mathbf{M}^{\boldsymbol{\theta}\delta} \\ \mathbf{M}^{\delta\boldsymbol{\theta}} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{\theta}} \\ \ddot{\boldsymbol{\eta}}^\delta \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Omega}^2 \end{bmatrix} \begin{bmatrix} \boldsymbol{\theta} \\ \boldsymbol{\eta}^\delta \end{bmatrix} = \begin{bmatrix} \mathbf{g}^a \\ \mathbf{0} \end{bmatrix} \tag{6.7}$$

$\boldsymbol{\Omega}^2 = \mathrm{diag}(\omega_i^2)$ is the diagonal matrix of the square eigenvalues. The mass matrix and $\boldsymbol{\Omega}^2$ depend nonlinearly on the configuration. This dependence is approximated by piecewise low-order polynomials. The configuration space is decomposed into boxes where the local polynomials are defined, in order to achieve a relative error below the tolerance $tol^e = 0.001$. The set of boxes are represented in Figure 6.15, and the variations of the model parameters are illustrated in Figure 6.16. The different orders of magnitude of the equivalent masses come from the different normalization of the rigid and flexible modes, as discussed in Figure 6.14.

The relation between the end effector position $\mathbf{x}^e$ and the modal coordinates is:

$$\mathbf{x}^e = \boldsymbol{\rho}^e(\boldsymbol{\theta}) + \boldsymbol{\Psi}^{e\delta}(\boldsymbol{\theta})\ \boldsymbol{\eta}^\delta \tag{6.8}$$

where $\boldsymbol{\rho}^e$ is the nonlinear kinematic transformation of the rigid mechanism, and $\boldsymbol{\Psi}^{e\delta}$ is the 2×2 matrix of the flexible modes at the end effector. At the velocity level, we have

$$\dot{\mathbf{x}}^e = \begin{bmatrix} \boldsymbol{\Psi}^{e\boldsymbol{\theta}} & \boldsymbol{\Psi}^{e\delta} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \dot{\boldsymbol{\eta}}^\delta \end{bmatrix} = \boldsymbol{\Psi}^{e\boldsymbol{\eta}}(\boldsymbol{\theta})\ \dot{\boldsymbol{\eta}} \tag{6.9}$$

$\boldsymbol{\Psi}^{e\boldsymbol{\theta}}$ is the 2×2 matrix of rigid modes at the effector. In this equation, the nonlinear contribution $\dot{\boldsymbol{\Psi}}^{e\boldsymbol{\eta}}(\boldsymbol{\theta})\ \boldsymbol{\eta}^\delta$ has been neglected under the assumption of small deformations.

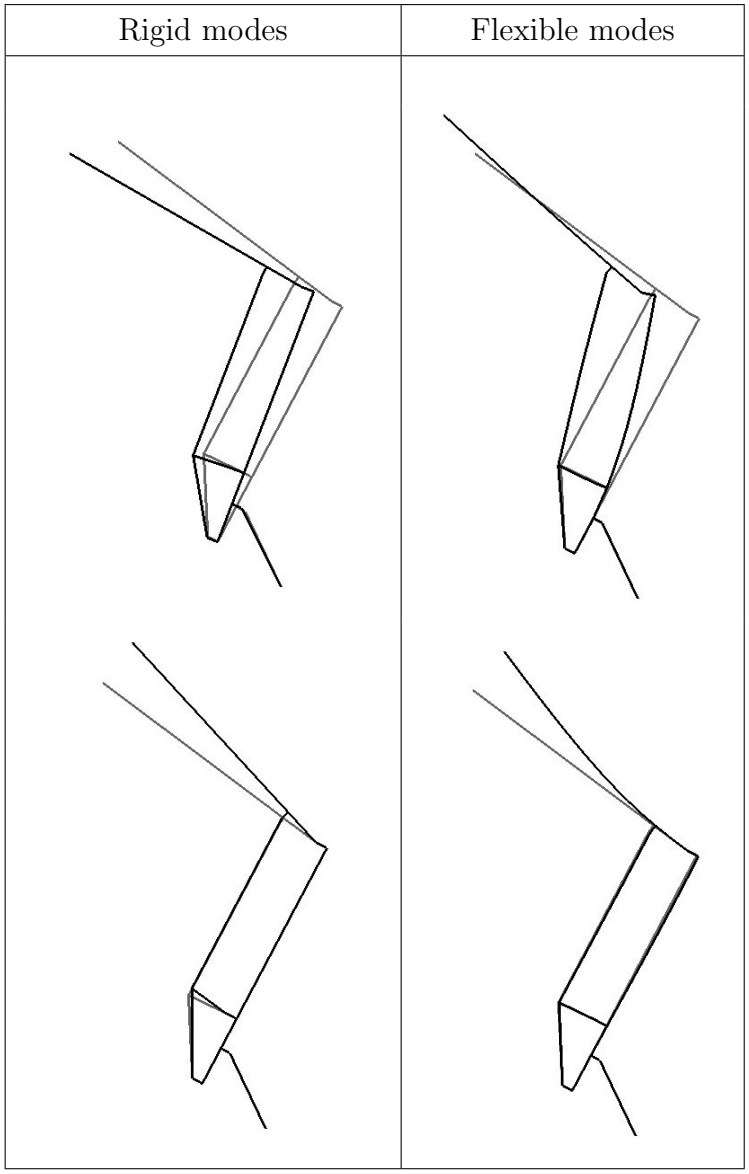


Figure 6.14: Mode shapes, home configuration. The amplitudes of the modes are $\delta\theta_1 = 0.05$ and $\delta\theta_2 = 0.08$, $\eta_1^\delta = 2.8$, and $\eta_2^\delta = 1.45$. Those different orders of magnitude come from the different normalization of the modal cordinates: the rigid modes are associated with unitary actuator displacements, whereas the flexible modes are normalized with respect to the mass matrix.

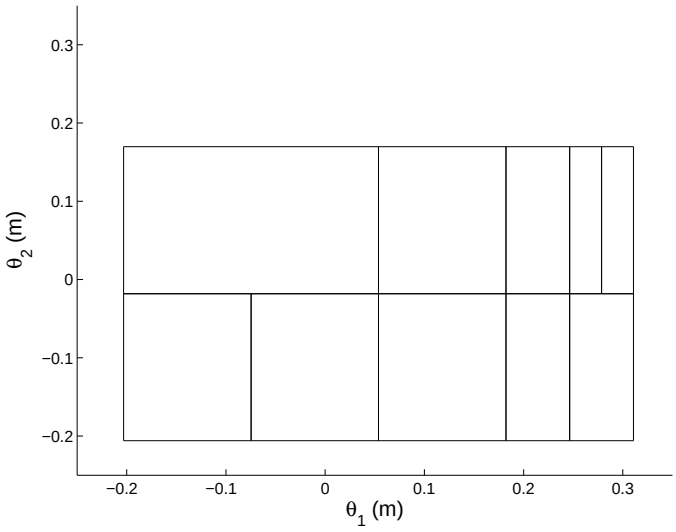


Figure 6.15: Configuration space discretization.

6.3 Control design

Most control algorithms tested on Ralf in the past were relying on a two-link assumed-mode model. Our modeling approach is more systematic and general, and it allows to account for the exact topology of the mechanism. In this section, the reduced-order model is exploited for the elaboration of a control strategy.

First, let us summarize previous researches on the control of Ralf. For a more general review of the literature on the control of flexible mechanisms, we refer to section 2.5. Lee [Lee90] complemented a PID joint controller with a decentralized strain feedback, in order to damp the vibration modes and to improve the performances of Ralf. Obergfell [Obe98] exploited the concept of output redefinition, reinterpreted in terms of combined positive deflection feedback with PD joint control. He developed an independent controller for each link/joint of the serial manipulator. The deflections were measured by optical sensors. In order to compensate for static deflections, an end-point position feedback was added as an external loop on the first controller. The absolute position was measured with a landmark tracking system, using a machine vision system. Even though good performances were obtained by Obergfell, the sensors are rather sophisticated, and their implementation imposes restrictive operational and design constraints.

Alternatively, we consider an accelerometer feedback, whose hardware imple-

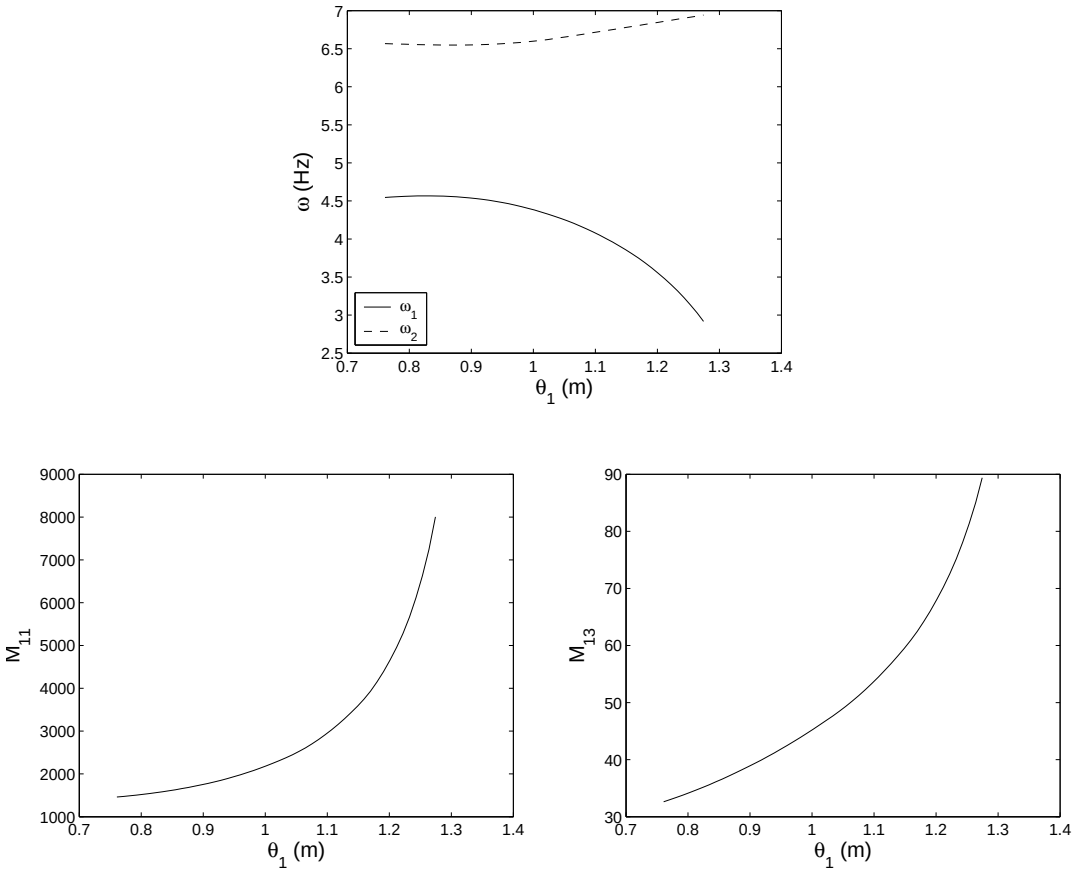


Figure 6.16: Variations of the model parameters along a diagonal line in the configuration space from $\underline{\boldsymbol{\theta}}$ (lower left corner) to $\overline{\boldsymbol{\theta}}$ (upper right corner). The variations of the natural frequencies are in the interval $[3,\ 7]\ Hz$. We recall that the modal masses of the flexible modes equals 1, as a consequence of their normalization.

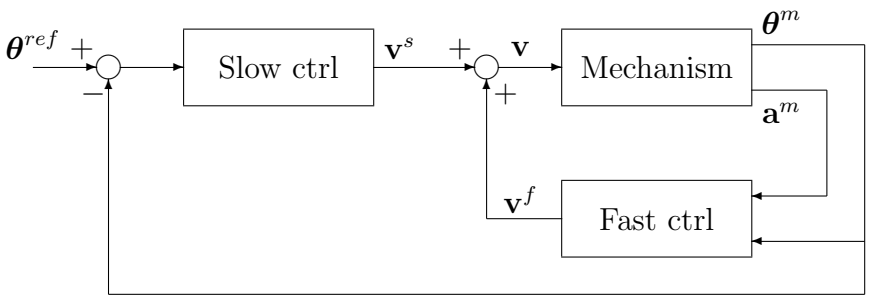


Figure 6.17: Two-time-scale control.

mentation is fairly simple and convenient. In particular, we investigate a composite *two-time-scale control strategy* (see Figure 6.17), which involves:

- a slow controller based on a feedback of the collocated measurements, responsible for the trajectory tracking,
- a fast controller, which relies on both collocated measurements and tip acceleration measurements, in order to increase the damping in the flexible modes.

The time-scale separation is a necessary condition to avoid interferences between the two controllers; in our case, we shall see that additional filters need to be implemented in order to fulfill this requirement.

In the literature, numerous authors applied two-time-scale control schemes for flexible mechanisms. Singular perturbation theory formulates a dynamic model in terms of a slow and a fast subsystem, which naturally conducts to the design of a composite controller [SB88, MC95, GLL98, MPK00, GS00, SV01, CYC02]. The HAC/LAC strategy [Pre97, VBS02, Sym04] is another pragmatic dual approach. The Low-Authority Controller (LAC) refers to the active damping, which is effective near the resonances, and which modifies the poles of the system only slightly. The High-Authority Controller (HAC) refers to the motion controller which dominates the low-frequency dynamics.

Our contribution especially focuses on the fast active damping controller; before a more detailed investigation, let us briefly describe the design of the slow controller.

6.3.1 Slow controller

In a serial robot, the actuated dofs are located at the joints, and roboticists are familiar with *decentralized control schemes*: each joint is controlled independently, using collocated feedback of the joint angle or position. This technique is opposed to the *tip-tracking strategy*, where one feeds back measurements or estimates of the tip position.

The slow controller presented here is a conventional joint-tracking controller, illustrated in Figure 6.18:

$$v_i^s = G_i^s \left(\theta_i^{ref} - \theta_i^m \right), \qquad i = 1, 2 \tag{6.10}$$

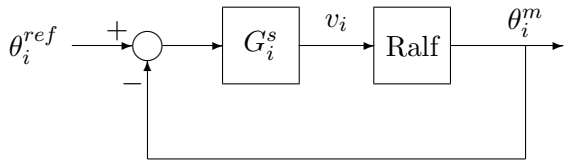


Figure 6.18: Joint-tracking controller for actuator i ($i = 1, 2$).

The gains G_i^s have to satisfy a double requirement:

- the joint-tracking control should be stable and performant,
- the bandwidth of the slow controller is limited by the natural frequencies of vibration, and it should be separated from the bandwidth of the fast controller.

At home configuration, the experimental transfer functions of the open-loop system are plotted in Figure 6.12. In order to guarantee sufficient phase margin near the resonance, the value of the gain is limited to 52 dB for both joints, so that

$$G_1^s = G_2^s = 400 \tag{6.11}$$

If one accepts the velocity source actuator model, the open-loop system is an integrator with gain G_i^{hydr}, and the slow controller leads to the closed-loop bandwidth:

$$\begin{aligned} \omega_i^s &= G_i^{hydr}\, G_i^s \\ \Rightarrow \quad \omega_1^s &= 9.4\ rad/s \quad (1.5\ Hz), \\ \Rightarrow \quad \omega_2^s &= 10.6\ rad/s \quad (1.7\ Hz). \end{aligned} \tag{6.12}$$

A parametric analysis in the configuration space has shown that those gain values lead to a stable behavior for any configuration of the mechanism. This has been confirmed experimentally.

When applying a conventional joint motion control which ignores flexibility, a rule-of-thumb is to restrict its bandwidth below the half [VBSN+01] or the third [Boo93] of the lowest natural frequency ω_1 of the mechanical system. In our case, ω_1 is about 4 Hz (see Figure 6.16), and the ratio ω^s/ω_1 is about 1/3. As we shall see, this ratio is sufficiently low to limit the interferences with the fast controller.

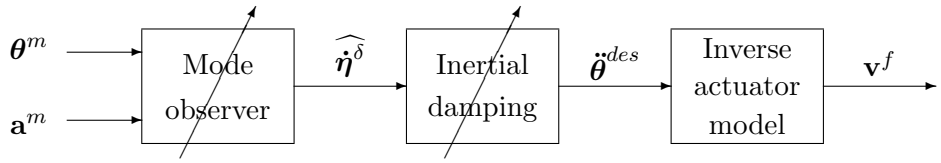


Figure 6.19: Fast controller. The arrows indicate a model-based scheduling according to the configuration changes.

6.3.2 Fast controller

The dynamics of the mechanism is configuration dependent. However, when designing the fast controller, the time-scale separation assumption allows to consider the nonlinear configuration changes as *quasi-static variations*. Concepts from linear system dynamics can thus be exploited to build a linear controller, whose parameters may be configuration dependent.

Thanks to the reduction procedure, the dynamic model of the mechanism (6.7) is in linear parameter-varying format, and it can be directly exploited for the design of such a fast controller. In this model, the action of the actuator forces $\mathbf{g}^a$ on the flexible modes comes indirectly from the inertial coupling. Therefore, in order to control the flexible modes, our idea is to excite the rigid modes so that the inertia forces produce a damping effect. This approach is equivalent to the inertial damping method developed for macro/micro-manipulators [Geo02].

The fast controller is composed of three stages illustrated in Figure 6.19:

- the *mode observer* extracts the rates of the flexible modes $\widehat{\dot{\boldsymbol{\eta}}^\delta}$ from the measurements,

- the *inertial damping control law* defines the desired accelerations of the rigid modes $\ddot{\boldsymbol{\theta}}^{des}$ that would produce appropriate inertial forces,

- the *inverse actuator model* establishes the voltages $\mathbf{v}^f$ to be applied to the actuators.

In the following, the inertial damping control law is first presented; the observer and the inverse actuator model will be described afterwards.

Inertial damping

In order to analyze the possible damping effect of the coupling inertia forces, the dynamic equation associated with the flexible modes is extracted from the reduced-order model (6.7):

$$\ddot{\boldsymbol{\eta}}^{\delta} + \boldsymbol{\Omega}^2 \, \boldsymbol{\eta}^{\delta} = -\mathbf{M}^{\delta\boldsymbol{\theta}} \, \ddot{\boldsymbol{\theta}} \tag{6.13}$$

The inertial damping consists in controlling the rigid variables $\boldsymbol{\theta}$ in order to produce a velocity feedback. Indeed, if $\ddot{\boldsymbol{\theta}}$ satisfies:

$$-\mathbf{M}^{\delta\boldsymbol{\theta}} \, \ddot{\boldsymbol{\theta}} = -\mathbf{G}^{f} \, \dot{\boldsymbol{\eta}}^{\delta} \tag{6.14}$$

the closed-loop equation of motion becomes:

$$\ddot{\boldsymbol{\eta}}^{\delta} + \mathbf{G}^{f} \, \dot{\boldsymbol{\eta}}^{\delta} + \boldsymbol{\Omega}^2 \, \boldsymbol{\eta}^{\delta} = \mathbf{0} \tag{6.15}$$

and $\mathbf{G}^{f} \, \dot{\boldsymbol{\eta}}^{\delta}$ has the clear interpretation of a damping term, provided that the gain matrix $\mathbf{G}^{f}$ is positive definite. The stability of the closed-loop system is thereby guaranteed. It is convenient to define the gain matrix with respect to the desired modal damping ratios ξ_i:

$$\mathbf{G}^{f} = diag(2 \, \xi_i \, \omega_i) \tag{6.16}$$

As will be motivated in the root locus analysis (section 6.4.2), our design is based on the choice

$$\xi_1 = \xi_2 = 0.1 \tag{6.17}$$

In order to generate this damping effect, the subsequent problem is to *control the rigid modes* so that equation (6.14) is satisfied. If the number of rigid modes equals the number of flexible modes, and if the inertia coupling matrix $\mathbf{M}^{\delta\boldsymbol{\theta}}$ is not singular, this equation can be inverted, leading to the definition of the *desired acceleration of the rigid modes* $\ddot{\boldsymbol{\theta}}^{des}$:

$$\boxed{\ddot{\boldsymbol{\theta}}^{des} = \left(\mathbf{M}^{\delta\boldsymbol{\theta}}\right)^{-1} \, \mathbf{G}^{f} \, \dot{\boldsymbol{\eta}}^{\delta}} \tag{6.18}$$

The matrix $\left(\mathbf{M}^{\delta\boldsymbol{\theta}}\right)^{-1}$ depends on the configuration. Recorded in the reduced-order model, it is represented by piecewise polynomial functions, which can be exported from the modeling environment to the real-time controller. $\left(\mathbf{M}^{\delta\boldsymbol{\theta}}\right)^{-1}$ is thus scheduled online, according to the slow configuration changes $\boldsymbol{\theta}^{m}$ detected by the linear sensors.

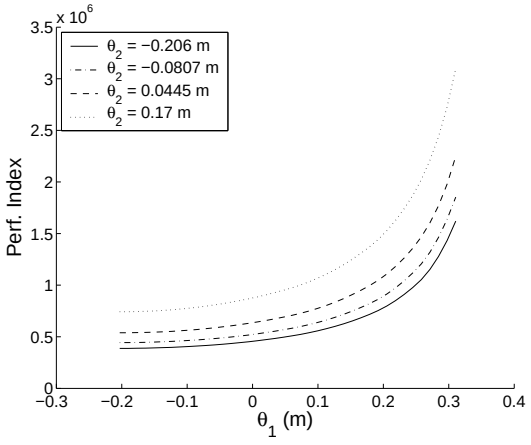


Figure 6.20: Performance index of the inertial damping control.

If the number of flexible modes to be controlled is lower than the number of rigid modes, this strategy is still applicable using the pseudo-inverse of the rectangular matrix $\mathbf{M}^{\delta\boldsymbol{\theta}}$.

It is necessary to check that no mechanical configuration is associated with an *inertial singularity*, *i.e.* a singularity of the coupling mass matrix $\mathbf{M}^{\delta\boldsymbol{\theta}}$. At an inertial singularity, it may be impossible to generate the required inertial forces. For this purpose, George [Geo02] proposed to analyze the variations of a performance index in the configuration space:

$$\text{Perf. Index} = \det\left(\mathbf{M}^{\delta\boldsymbol{\theta}^T}\,\mathbf{M}^{\delta\boldsymbol{\theta}}\right) \tag{6.19}$$

This definition is still applicable if the coupling matrix $\mathbf{M}^{\delta\boldsymbol{\theta}}$ is not square. Since the determinant is the product of the singular values, the higher the performance index, the farther from inertial singularities. In Figure 6.20, we observe that no singularity is present in the configuration space of Ralf.

Mode observer

In the inertial damping control law (6.18), $\ddot{\boldsymbol{\theta}}^{des}$ is computed from the modal velocities $\dot{\boldsymbol{\eta}}^{\delta}$, which are not directly available. The observer aims at reconstructing $\dot{\boldsymbol{\eta}}^{\delta}$ from the measurements $\boldsymbol{\theta}^m$ and $\mathbf{a}^m$. First, let us characterize the relation between the accelerometer signals and the modal coordinates.

The accelerometers measure the absolute accelerations in body axes, and those axes rotate during the motion. If $\mathbf{R}$ denotes the rotation matrix of the end

effector node, and $\mathbf{x}^e$ its position referred to an inertial frame, we have:

$$\ddot{\mathbf{x}}^e = \mathbf{R}\ \mathbf{a}^m \tag{6.20}$$

Under the assumption of small deformations, $\mathbf{R}$ only depends on the configuration $\boldsymbol{\theta}$, and not on the flexible modes. Thus, in the fast time-scale, the variations of $\mathbf{R}$ are quasi-static and the integration of (6.20) leads to

$$\dot{\mathbf{x}}^e = \int_0^t \mathbf{R}\ \mathbf{a}^m\ d\tau \simeq \mathbf{R} \int_0^t \mathbf{a}^m\ d\tau \tag{6.21}$$

Defining the integrated accelerometer signal $\mathbf{c}^{ma}$

$$\mathbf{c}^{ma} = \int_0^t \mathbf{a}^m\ d\tau \tag{6.22}$$

we obtain

$$\dot{\mathbf{x}}^e \simeq \mathbf{R}\ \mathbf{c}^{ma} \tag{6.23}$$

Inverting this expression, and introducing the modal amplitudes of the end effector displacements $\dot{\mathbf{x}}^e = \boldsymbol{\Psi}^{e\boldsymbol{\eta}}\ \dot{\boldsymbol{\eta}}$, we obtain:

$$\mathbf{c}^{ma} = \boldsymbol{\Psi}^{a\boldsymbol{\eta}}\ \dot{\boldsymbol{\eta}} \qquad \text{with} \qquad \boldsymbol{\Psi}^{a\boldsymbol{\eta}}(\boldsymbol{\theta}) = \mathbf{R}^T(\boldsymbol{\theta})\ \boldsymbol{\Psi}^{e\boldsymbol{\eta}}(\boldsymbol{\theta}) \tag{6.24}$$

The relation between the measurements and the modal coordinates is summarized by:

$$\begin{bmatrix} \dot{\boldsymbol{\theta}}^m \\ \mathbf{c}^{ma} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \boldsymbol{\Psi}^{a\boldsymbol{\theta}} & \boldsymbol{\Psi}^{a\delta} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \dot{\boldsymbol{\eta}}^\delta \end{bmatrix} \tag{6.25}$$

Since there are as many sensors as modal coordinates, this expression can be inverted, leading to a *static estimation formula* for the modal velocities:

$$\widehat{\dot{\boldsymbol{\eta}}^\delta} = \underbrace{\begin{bmatrix} -\left(\boldsymbol{\Psi}^{\mathbf{a}\delta}\right)^{-1} \boldsymbol{\Psi}^{\mathbf{a}\boldsymbol{\theta}} & \left(\boldsymbol{\Psi}^{\mathbf{a}\delta}\right)^{-1} \end{bmatrix}}_{=\boldsymbol{\Psi}^{\delta m}} \begin{bmatrix} \dot{\boldsymbol{\theta}}^m \\ \mathbf{c}^{ma} \end{bmatrix} \tag{6.26}$$

The estimation algorithm is illustrated in Figure 6.21, and it requires the 2×4 modal matrix $\boldsymbol{\Psi}^{\delta m}$ to be available online. The importance of the filters shall be discussed later on.

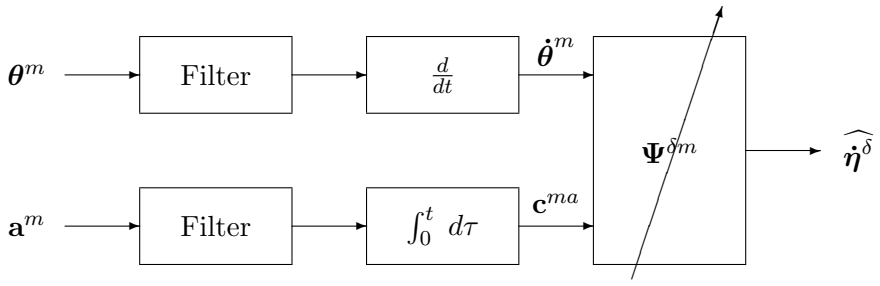


Figure 6.21: Observer. The signals of the sensors are first filtered, then numerically differentiated/integrated with respect to time, and finally, equation (6.26) is used to estimate the modal velocity.

Inverse actuator model

The velocity source actuator model (6.2) can be inverted, leading to the definition of the applied voltage:

$$v_i^f = \frac{1}{G_i^{hydr}} \int_0^t \ddot{\theta}_i^{des} \, d\tau, \qquad i = 1, 2 \tag{6.27}$$

In order to avoid any drift in the numerical integration, the integrator is implemented as a high-pass integrator, *i.e.* an integrator in series with a high-pass filter.

The construction of the inverse model from the first actuator model has been motivated by the direct connection that it defines between the input voltage v_i and the actuator motion θ_i. On the contrary, the second actuator model involves the actuator force, so that the input/output relation between v_i and θ_i is affected by the whole dynamics, in particular, by the mechanical resonances. The inversion of this model would be more complicated and sensitive to modeling error.

One may be afraid that the velocity source model is not satisfactory near the resonances, where the fast controller is expected to be performant. However, the root locus analysis presented in section 6.4.2 will assess the stabilization effect of the resulting control law.

6.3.3 Summary and first results

In essence, the inertial damping strategy defined by (6.18) is conflicting with the slow controller which tries to achieve $\boldsymbol{\theta} = \boldsymbol{\theta}^{ref}$. This paradox is resolved by

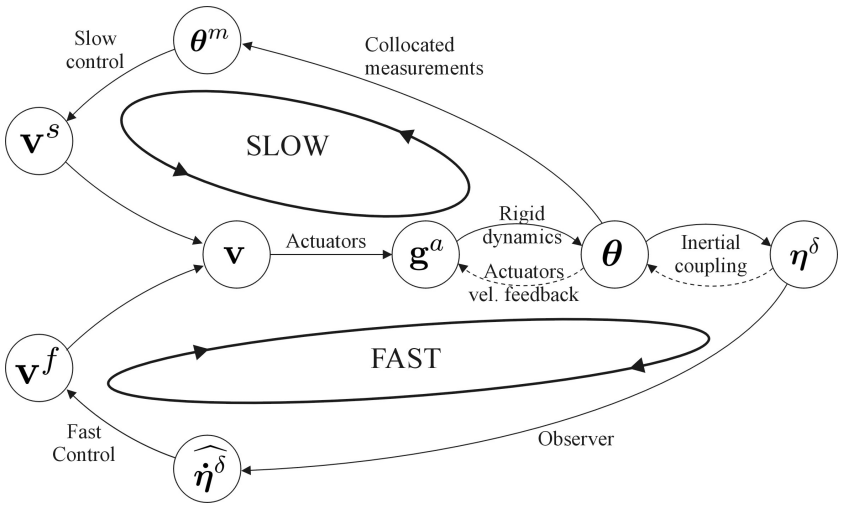


Figure 6.22: Signal-flow graph of the controlled mechanism: slow and fast control loops.

imposing that the inertial damping is only active at high-frequencies, near the resonances of the flexible modes $\boldsymbol{\eta}^\delta$.

In order to clarify this point, let us summarize the interactions of both controllers with the flexible mechanism. In Figure 6.22, the signal-flow graph illustrates the dynamics of the controlled mechanism; in particular, the slow and fast control loops are clearly represented. Let us briefly comment the graph. In the second actuator model (6.4), the actuator forces $\mathbf{g}^a$ are influenced by the applied voltage $\mathbf{v}$ and by the extension rate $\dot{\boldsymbol{\theta}}$. According to the mechanical model (6.7), these forces excite the rigid dynamics described by the actuator coordinates $\boldsymbol{\theta}$, which is coupled with the flexible dynamics described by the modal amplitudes $\boldsymbol{\eta}^\delta$. The joint-tracking controller is based on collocated measurements $\boldsymbol{\theta}^m$, whereas the fast controller feeds back an estimation of the flexible modal rates $\widehat{\dot{\boldsymbol{\eta}}^\delta}$.

The interaction between both control loops is limited under the time-scale separation assumption. Indeed, the closed-loop bandwidth of slow controller is limited to $\omega^s \simeq 1/3\ \omega_1$, whereas the fast controller is a feedback of the flexible modal coordinates, which are mainly excited near the resonances. At first sight, the time-scale separation seems to be satisfied, and good performances are expected from the composite controller.

However, we argue that various disturbances may be responsible for unde-

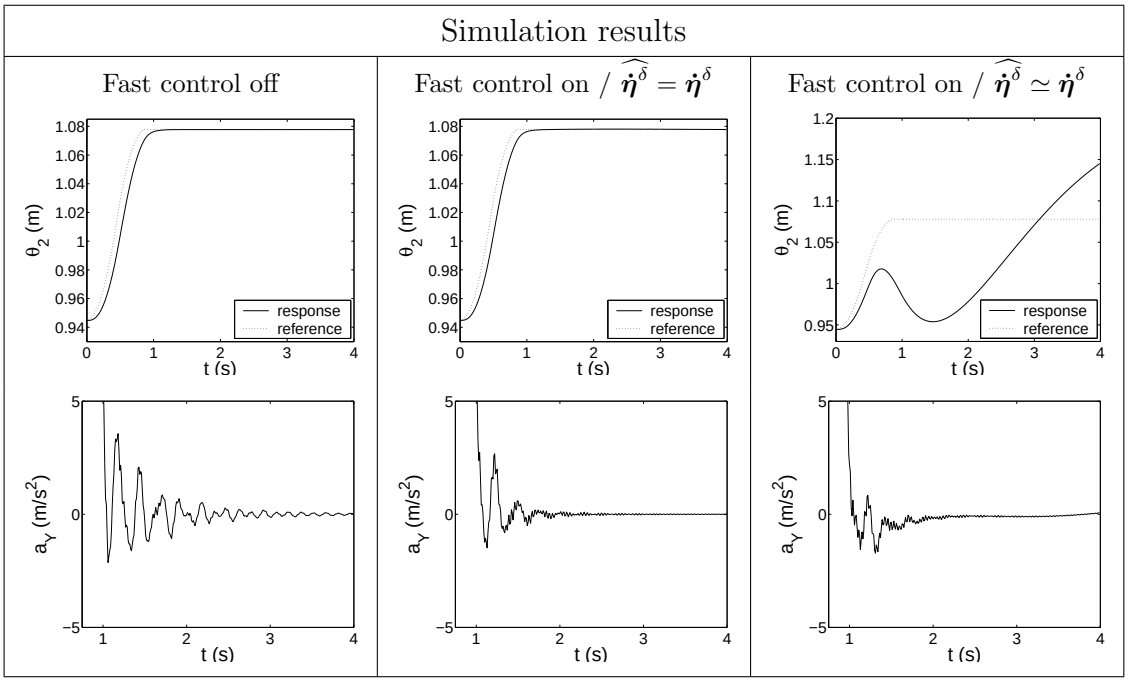


Figure 6.23: Point-to-point trajectory. Sensitivity to an observer error.

sirable interactions between the slow and the fast controller. To illustrate this, Figure 6.23 presents simulation results for a point-to-point trajectory (more details about the model will be given later). The left plot shows the slow controller working alone. Good tracking performances are observed for θ_2, but the tip vibrations decay slowly after reaching the final configuration. In the middle plot, the fast control law is activated, assuming a perfect estimation of the modal velocities. The improvement of the tip response is remarkable, and the joint-tracking performances are barely affected. However, when one accounts for the non-perfect estimation of $\dot{\boldsymbol{\eta}}^{\delta}$ due to the approximation in equation (6.23), the fast controller is still efficient, but the joint-tracking controller is completely disturbed. This is easily explained, since the approximation (6.23) is only justified in the fast time-scale. In the slow time-scale, a significant error is introduced and amplified by the inverse actuator model, which directly affects the slow controller.

It is possible to improve the observer in order to eliminate this particular source of error. However, in the experimental system, other disturbances may affect the observer at low-frequencies, such as the accelerometer errors. Therefore, there is a need to eliminate the low-frequency components induced in the fast

controller by appropriate filtering. The design of those filters is presented in the following section.

Filters design

The fast controller is expected to be active around the first two natural frequencies, between 3 Hz and 7 Hz. Both the lower and higher frequency disturbances should be attenuated. Two Butterworth *band-pass filters* are thus considered, whose properties are illustrated in Figure 6.24. Filter A is a 4th order filter, with corner frequencies at 1 and 25 Hz, whereas filter B is a 2nd order filter with corner frequencies at 1.6 and 16 Hz. Both filters have been designed in order to keep the phase shift below 20 deg in the interval [3, 7] Hz. Filter A has the advantage of a larger roll-off, leading to an efficient attenuation of low-frequency disturbances. However, its lower corner frequency is smaller than for filter B, so that its performances at intermediate frequencies, around $\omega^s = 1.5$ Hz, are less interesting. Both filters will be tested in simulations and experiments.

Two less critical filters can also be mentioned:

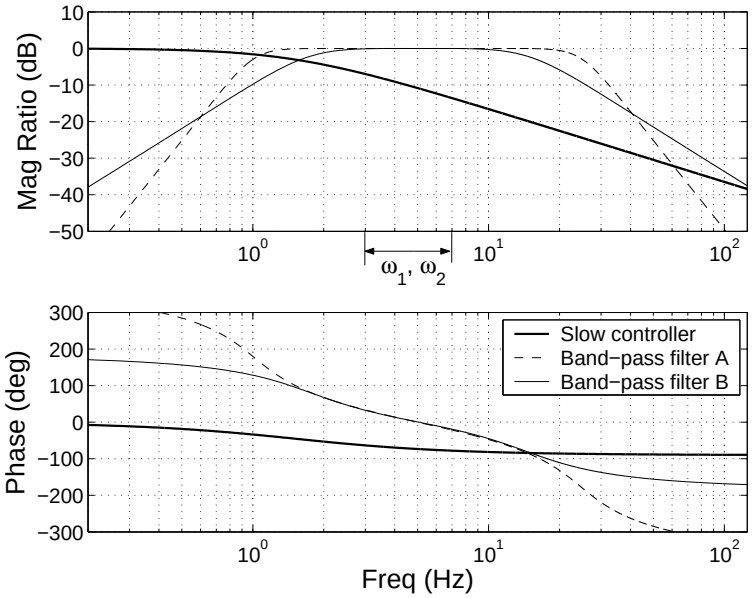
- an output high-pass filter associated with the inverse actuator model (4th order Butterworth filter, with corner frequency at 0.1 Hz),

- a low-pass filter for the slow controller in order to reduce the measurement noise (2nd order Butterworh filter, with corner frequency at 100 Hz).

6.4 Closed-loop system: simulations and experiments

In this section, the closed-loop system is simulated and tested experimentally. The dynamic model includes the model of the mechanism, of the actuator, and of the controller. As mentioned in Table 6.1 at the beginning of this chapter, different models of the mechanism are considered for different purposes:

- a reduced-order linear model is used for the construction of root loci, and the analysis of frequency responses,



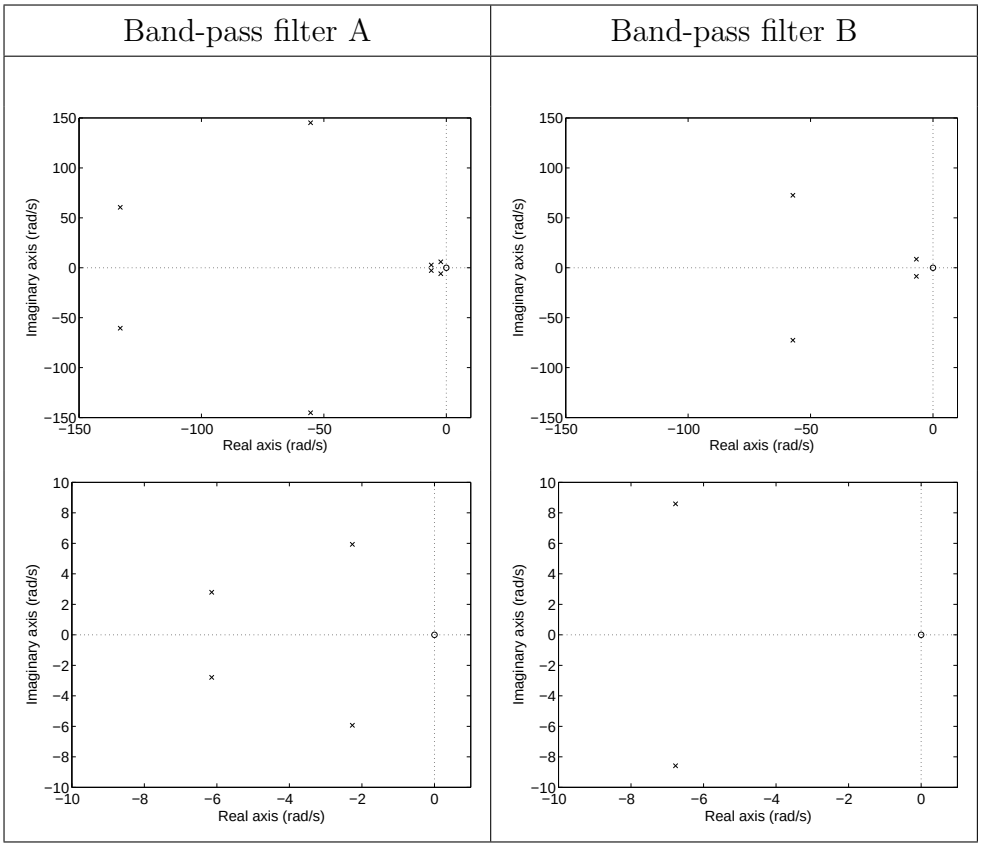


Figure 6.24: Band-pass filters at the inputs of the fast controller. In the Bode plot, the closed-loop transfer function of the slow controller is represented, assuming a velocity source actuator model (actuator 1).

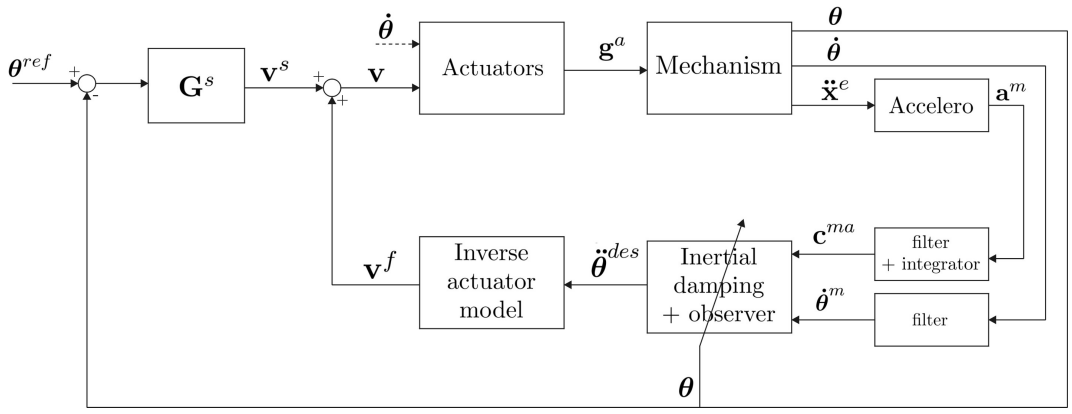


Figure 6.25: Model of the mechatronic system.

- a full-order nonlinear Finite Element model is considered for simulations in the time-domain, *i.e.* the response to a tip force, point-to-point trajectories and a square trajectory,

- a reduced-order nonlinear model is exploited to reduce the computational load associated with time-domain simulations.

Two categories of analyses are defined: small motion around the home configuration, and large amplitude trajectories in the configuration space. Before the presentation of the experimental and simulation results, the model of the mechatronic system is described.

6.4.1 Model of the mechatronic system

In the experimental setup, the control law is implemented on a digital computer, with a sampling rate of 1 kHz. The sampling effect has thus little effects on the dynamic behavior in the frequency range of interest; we assume that it can be neglected in the model.

Figure 6.25 represents the model of the mechatronic system. Each block is implemented according to the formalism defined in chapter 4:

- *Mechanism.* Different models of the mechanism are used for the different analyses, as mentioned above.

- *Actuators.* The actuator model accounts for the intrinsic velocity feedback defined by equation (6.4).

- *Accelerometers.* In the Finite Element model, the tip accelerations $\ddot{\mathbf{x}}^e$ are available in inertial axes[1]. In body axes, the accelerometer signals $\mathbf{a}^m$ are computed by inversion of the formula (6.20) [2].

- *Filters and integrator.* The filters are continuous linear state-space systems.

- *Inertial damping + observer.* A nonlinear algebraic equation is defined by combination of (6.18) and (6.26):

$$\ddot{\boldsymbol{\theta}}^{des} = \underbrace{\left(\mathbf{M}^{\delta\boldsymbol{\theta}}\right)^{-1} \mathbf{G}^f \boldsymbol{\Psi}^{\delta m}}_{=\widehat{\mathbf{G}^f}(\boldsymbol{\theta})} \begin{bmatrix} \dot{\boldsymbol{\theta}}^m \\ \mathbf{c}^{ma} \end{bmatrix} \tag{6.28}$$

  $\widehat{\mathbf{G}^f}(\boldsymbol{\theta})$ is implemented in the controller as a C-function, which is directly reused in the simulation model.

- *Inverse actuator model.* Formula (6.27) is combined with a high-pass filter.

6.4.2 Dynamic behavior at home configuration

First of all, the closed-loop system is analyzed when Ralf is at home configuration: root loci, bode plots, and step responses will be successively analyzed. The *linearized model* of the mechatronic system is sufficient, and the reduced-order mechanical model is exploited for analysis in the frequency-domain. The simulation of the step response in the time-domain is based on the full Finite Element model, according to the methodology proposed in chapter 4.

For fixed boundary conditions at the level of the actuators, the model predicts the natural frequencies of the flexible modes at home configuration:

$$\omega_1 = 4.4 \quad Hz, \qquad \omega_2 = 6.7 \quad Hz \tag{6.29}$$

In the model, the flexible modes are characterized by a small structural damping (around 1 %). Obviously, in the closed-loop system, those poles are affected by the behavior of the control system.

[1] According to Remark 4.2 page 75, an "observer equation" is added for the accurate estimation of $\ddot{\mathbf{x}}^e$.

[2] Due to rather slow variations of the rotation matrix $\mathbf{R}$ with respect to the time-step ($h = 0.01\ s$), this formula is not strongly nonlinear, and does not represent a problem for the numerical integration scheme (see Remark 4.4, page 84)

Root locus

The root loci for increasing gains of the fast controller ξ_i $(i = 1, 2)$ are presented in Figure 6.26. Due to interactions with the actuators and the slow controller, the flexible modes have initially a significant non-negligible damping. In the absence of band-pass filtering, the vibration controller leads to significant damping improvements. Considering the position of the poles for $\xi_i = 0.1$, there is room for further improvements. However, in the presence of filtering, a smaller active damping can be reached for the first flexible mode. For large gain values, unstable behaviors are predicted, and the value $\xi_i = 0.1$ appears as a reasonable choice.

Frequency response

The tracking performances of the composite controller can be analyzed in the transfer functions between the reference inputs and the dynamic responses. Generally, a joint-tracking controller tries to optimize the responses of the actuator dofs $\boldsymbol{\theta}$, but in practice one is more interested in the positioning of the end effector. The impact of the fast controller on both types of transfer functions is illustrated in Figure 6.27.

First, let us consider the predictions of the model when the fast controller is turned off. In the actuator transfer function, the corner frequency is about 1.5 Hz, and the roll-off is -20 dB/decade. 4.4 Hz corresponds to the first resonance of the mechanism with clamped actuators. At this frequency, any motion of the actuator is amplified in the whole structure and requires a huge amount of energy. This phenomenon appears as an anti-resonance in the collocated transfer function. In the tip transfer function, the flexible modes are excited at the resonance despite the roll-off of the joint-tracking controller.

When the fast controller is turned on, the actuator transfer function is modified around the resonance, where the inertial damping is active. The benefits mostly appear in the end effector transfer function, where the resonance peaks are efficiently attenuated. Due to phase lags, the tracking behavior is still not ideal around the resonance, but the system is now isolated from disturbances in this frequency range. This clearly illustrates the objective of the composite control law: performances at low frequencies, and stabilization at the resonances.

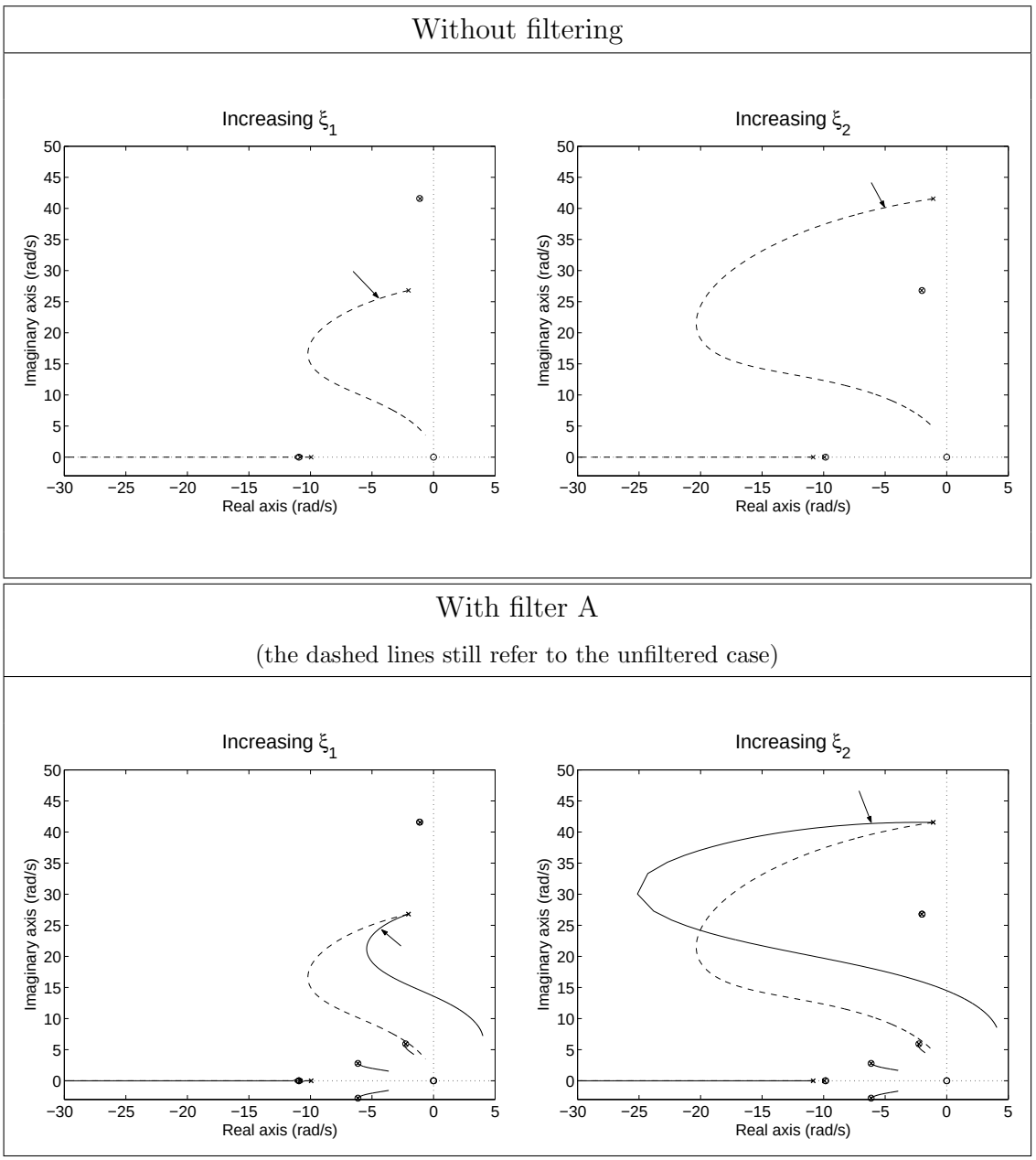


Figure 6.26: Root loci. The arrows point to $\xi_i = 0.1$. In the second plot, remote poles and loci associated with the higher corner frequency of the filters exist out of the axes scales.

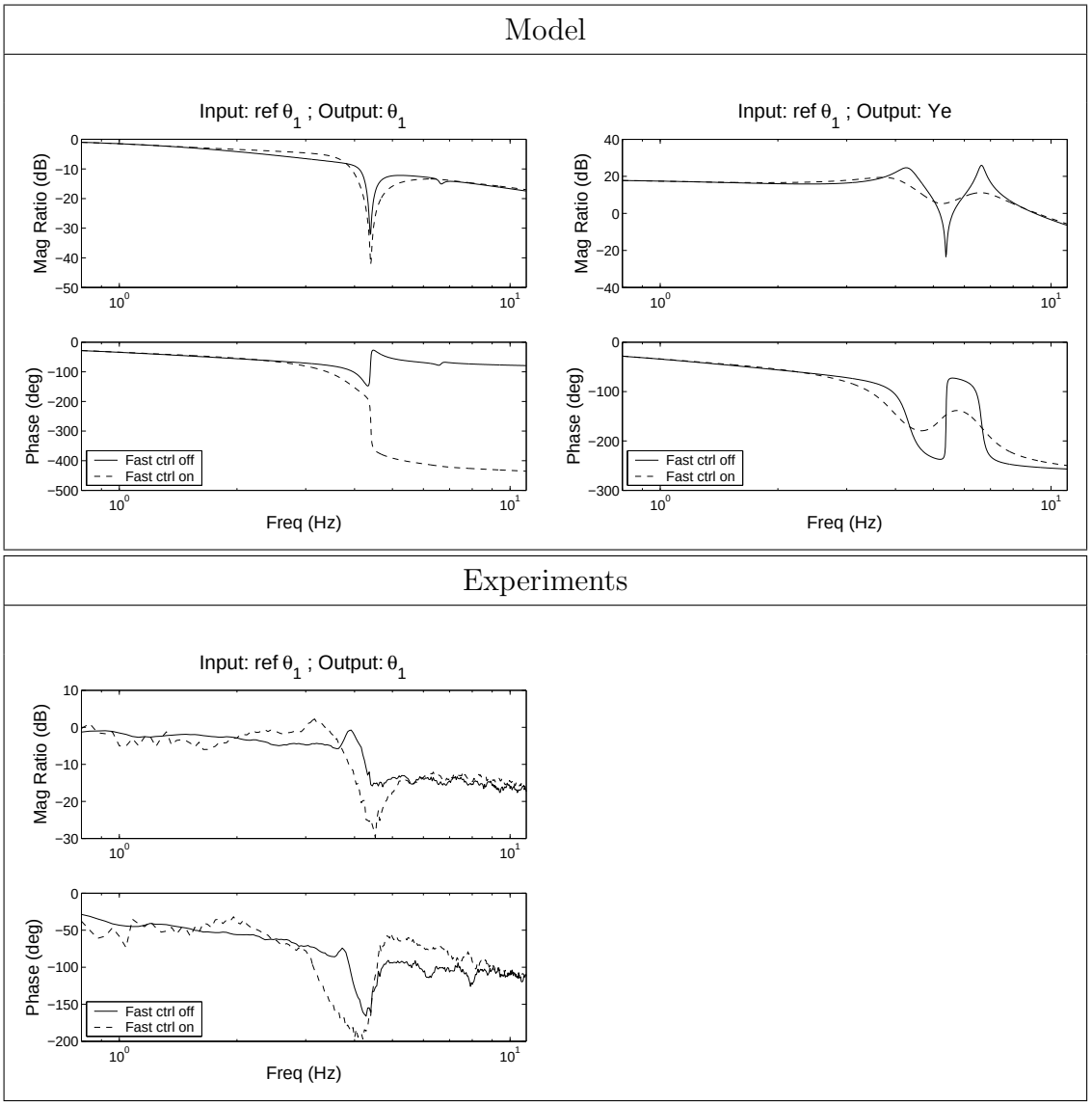


Figure 6.27: Closed-loop transfer function (home configuration). Y^e represents the displacements in body axes, in the direction of the accelerometer measurement a_Y^m.

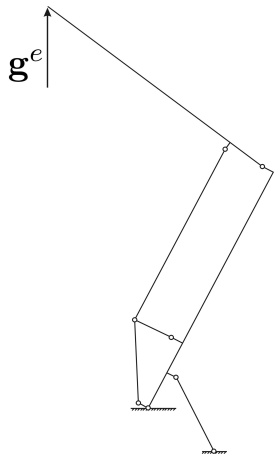


Figure 6.28: Step force applied to the tip of Ralf (home configuration).

An experimental transfer function is also represented; at least from a qualitative point of view, the predictions of the model are confirmed.

Response to a tip force

The actively damped system should be less sensitive to external disturbances. In order to demonstrate this capability, the response of the system is analyzed when an external force is suddenly applied at the tip, as illustrated in Figure 6.28. The step of force is exerted at time $t = 0\ s$, with an amplitude $g^e = 89\ N$.

First, the simulation of this problem is considered. The full Finite Element model is used for the modeling of the mechanism, and the control system is modeled using the block diagram formalism developed in chapter 4. The Hilber-Hughes-Taylor time-integration scheme is exploited with a time-step $h = 0.01\ s$ and a numerical damping $\alpha_f = 0.02$. The step of force is a discontinuous phenomenon, and consistent initial conditions have been carefully computed.

This problem has also been analyzed experimentally. The step of force is emulated using a payload fixed to the tip of Ralf with a cable: the instantaneous cutting of the cable gives an equivalent upright force. In practice, the cutting operation is replaced by a trigger mechanism.

In Figure 6.29, the correspondence between simulation and experimental results is impressive. The limited resolution of the measured signal θ_2^m appears in the experimental data. A tiny shift is noticed in the static value of θ_2^m; it is caused

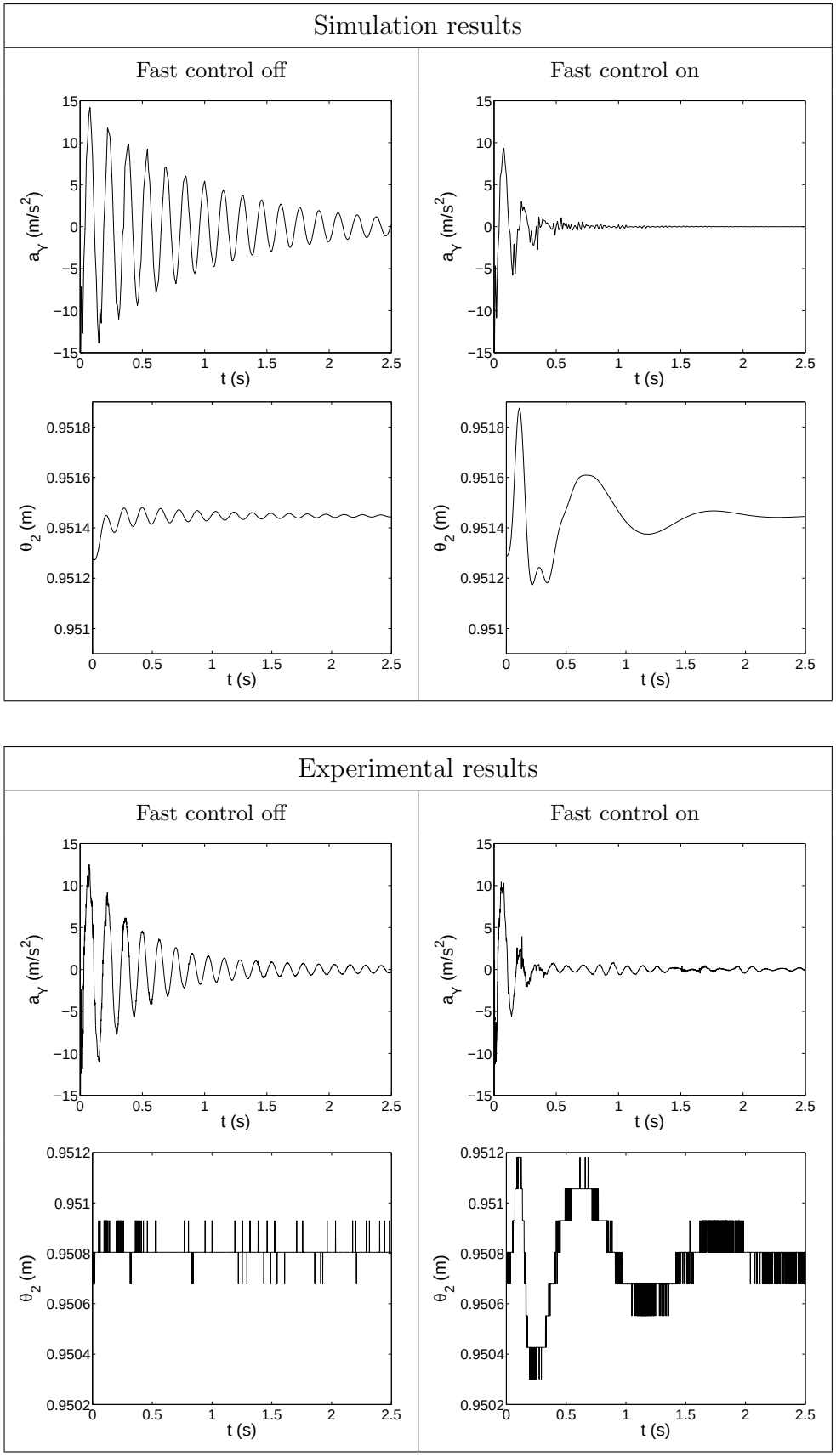


Figure 6.29: Tip disturbance. Filter A is used in the control algorithm.

by the offsets of the hydraulic valves.

When the fast controller is not active, the actuator remains almost motionless, and the tip response is dominated by the *second mode of vibration*. In the model, the main source of damping comes from the actuators. The decay of the experimental response is faster, which might be due to unmodeled phenomena in the joints, such as friction effects.

The active vibration controller efficiently reduces the tip vibrations. This performance is obtained at the price of a perturbed actuator motion, dominated by a pole associated with the band-pass filter (in this case, filter A, with corner frequency at 1 Hz). In the next section, improvements will be discussed for the attenuation of this phenomenon.

6.4.3 Trajectories in the configuration space

After the analysis of the controller performances around the home configuration, the next step is to consider large amplitude motions, where several nonlinear effects may arise, due to configuration changes, to the scheduled control strategy, and to observer errors. The composite controller should also cope with the dual task trajectory control/active damping.

When θ_1 increases, the system gets closer to a singular configuration, where actuator 1 is aligned with the first link[3]. For instance, in Figure 6.2, the slope $\frac{\partial \alpha_1}{\partial \theta_1}$ increases with θ_1. As a consequence, any motion of the actuator is amplified by the mechanism, and for a desired transmitted power, higher forces g_1^a are required. The controllability measures the ability of the actuator forces to excite the system; it is thus optimal when actuator 1 is orthogonal to the first link[4] and it vanishes at the actuator singularity.

Therefore, the implementation of the control law is analyzed in a restricted part of the configuration space $\theta_1 < 1.12\ m$ ($\alpha_1 < 84\ deg$), where the controllability is sufficient for the design of an efficient controller. The trajectories follow straight paths defined in the space of joint angles (Figure 6.30). Any of these motions involves both actuators 1 and 2. A point-to-point motion and a square

[3]Rigorously, at the singularity, actuator 1 is orthogonal to the virtual displacement of its attachment point.

[4]Rigorously, the controllability is optimal when actuator 1 is parallel to the virtual displacement of its attachment point.

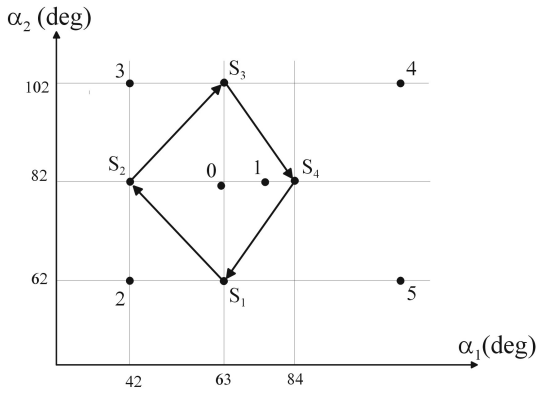


Figure 6.30: Point-to-point motions in the configuration space.

trajectory will be successively considered.

The mechatronic system is modeled and simulated as in the previous case (full Finite Element model, block-diagram model of the control system, Hilber-Hughes-Taylor algorithm, $h = 0.01\ s$, $\alpha_f = 0.02$).

Point-to-point motion

A smooth point-to-point trajectory from configuration S_4 to configuration S_1 is analyzed. The trajectory $\boldsymbol{\theta}^{ref}(t)$ follows bang-bang acceleration profiles, *i.e.* $\ddot{\boldsymbol{\theta}}^{ref}$ is piecewise constant and the velocity $\dot{\boldsymbol{\theta}}^{ref}$ is limited. At the end configuration ($t = 0.85\ s$), the transition is not perfectly smooth in order to emulate possible high-frequency disturbances.

Simulation and experimental results are compared in Figures 6.31. In the *simulation results*, three subcases are studied. When the vibration controller is turned off, the slow controller is able to track efficiently the trajectory, with a small delay. However, tip vibrations continue for a long time after the end of the trajectory. Those vibrations are dominated by the *first flexible mode*. Without filtering, the fast controller leads to an important reduction of those transient vibrations, but the tracking controller is disturbed in an unacceptable way, as discussed earlier. Filter A leads to a better trade-off between vibration control and trajectory tracking; the slow oscillations at the actuator level are dominated by the poles of the filter.

Qualitatively, the *experiments* agree with those observations (the non-filtered

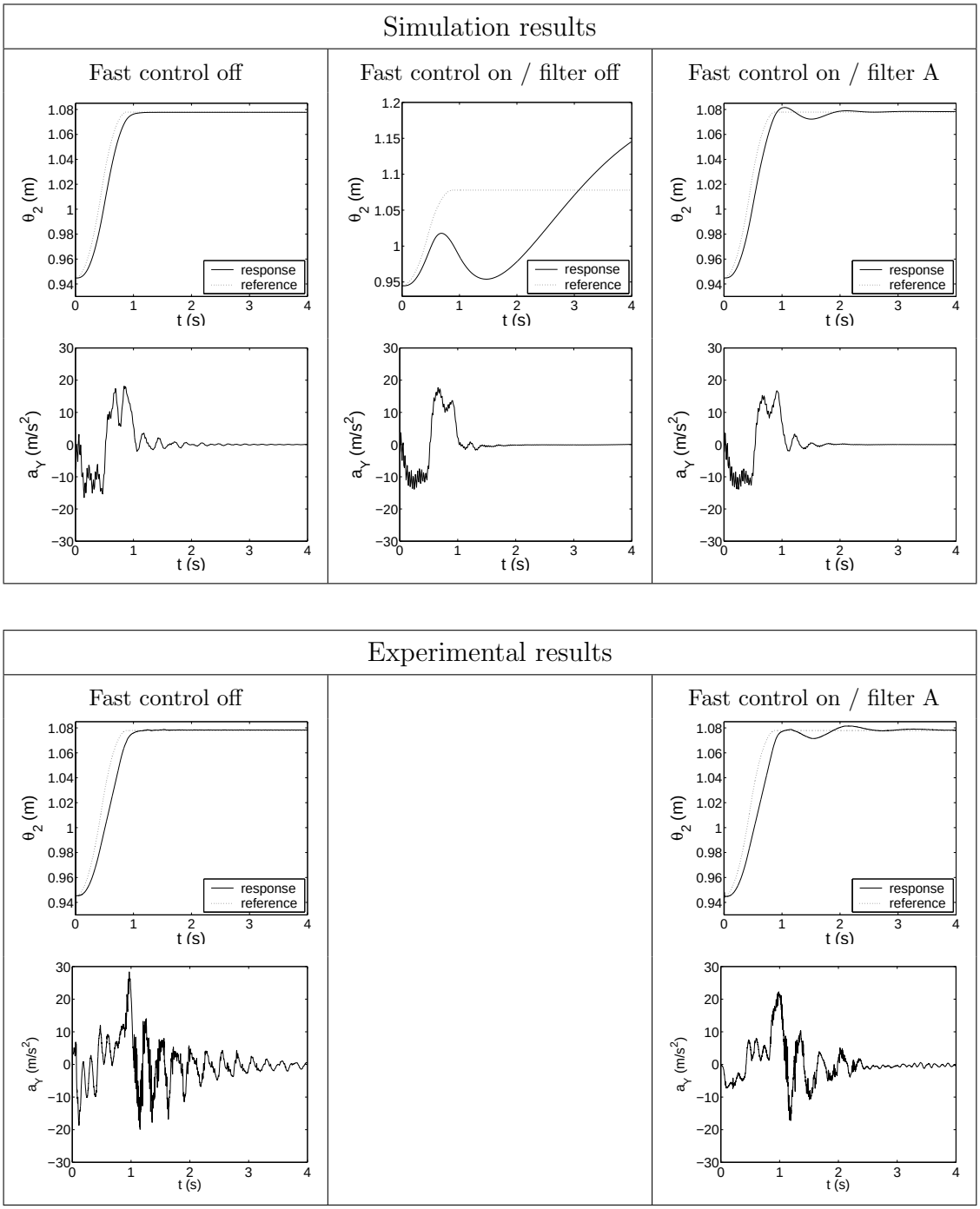


Figure 6.31: Point-to-point trajectory.

fast controller has not been tested, fearing unpredictable motions of the mechanism). However, the amplitude level of the transient vibrations after the motion is higher than the prediction of the simulation, even when the fast controller is disabled.

Part of this discrepancy may be attributed to unmodeled phenomena and disturbances. For instance, *clearance effects in the joints* may excite the flexible motion at the end of the trajectory. In the tip-force experiment, the predictions of the model were far closer to the experimental results, which can be explained as follows. The response to a tip force is dominated by the second mode of vibration, which is almost a pure bending mode of the second link, as represented in Figure 6.14. Thus, the joints do not participate in the motion, and no clearance phenomenon is observed. The situation is quite different for the point-to-point motion, since the first mode of vibration is then excited.

Another explanation may come from the *non-perfect actuator model* around the first natural frequency. Indeed, in Figure 6.12, it was difficult to make a decision whether the sensor measurements and the actuators were collocated or not. For our developments, we assumed collocated measurements, but, *a posteriori*, an additional simulation has been realized for the non-collocated case. The comparison is given in Figure 6.32. For the noncollocated case, without active vibration control, the vibrations detected by the linear sensors have a destabilizing effect on the slow controller. When the vibration controller is activated, the efficiency of the active damping is overestimated by the model. We may conclude that the true reality is between the collocated and the non-collocated situations. Experimental data could be exploited for the optimization of an "intermediate" model. In the following, this identification problem is left aside, and the simulation results rely on the assumption of collocated measurements.

Figure 6.33 focuses on the behavior of the system at the end of the trajectory, when the final configuration has been reached. The damping performances of the fast controller appear more clearly. Filter B turns out to be more appropriate in this case: the slow motion of the actuator is more efficiently damped.

In order to improve the computational efficiency of the simulation algorithm, the full Finite Element model of the mechanism can be replaced by the nonlinear reduced model. The reduced model describes the dynamics in terms of modal coordinates. The tip position is connected with the modal amplitudes by the

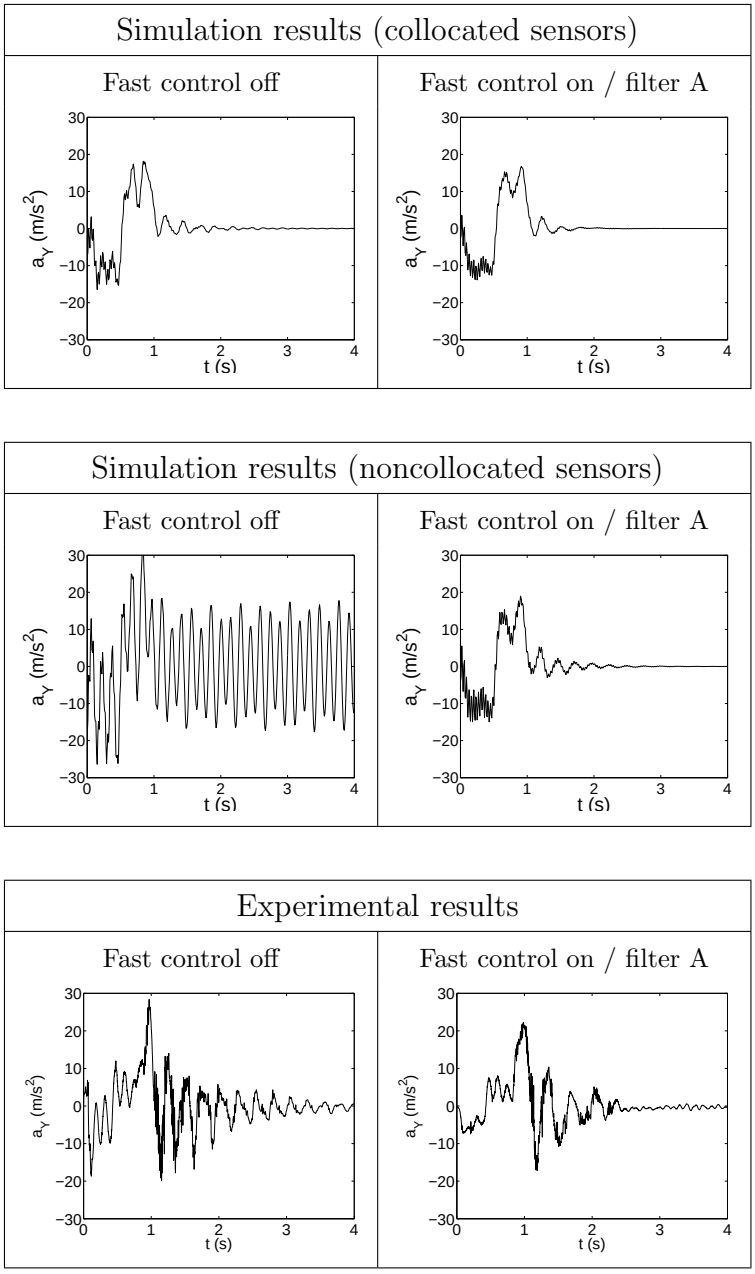


Figure 6.32: Comparison: collocated / noncollocated measurements.

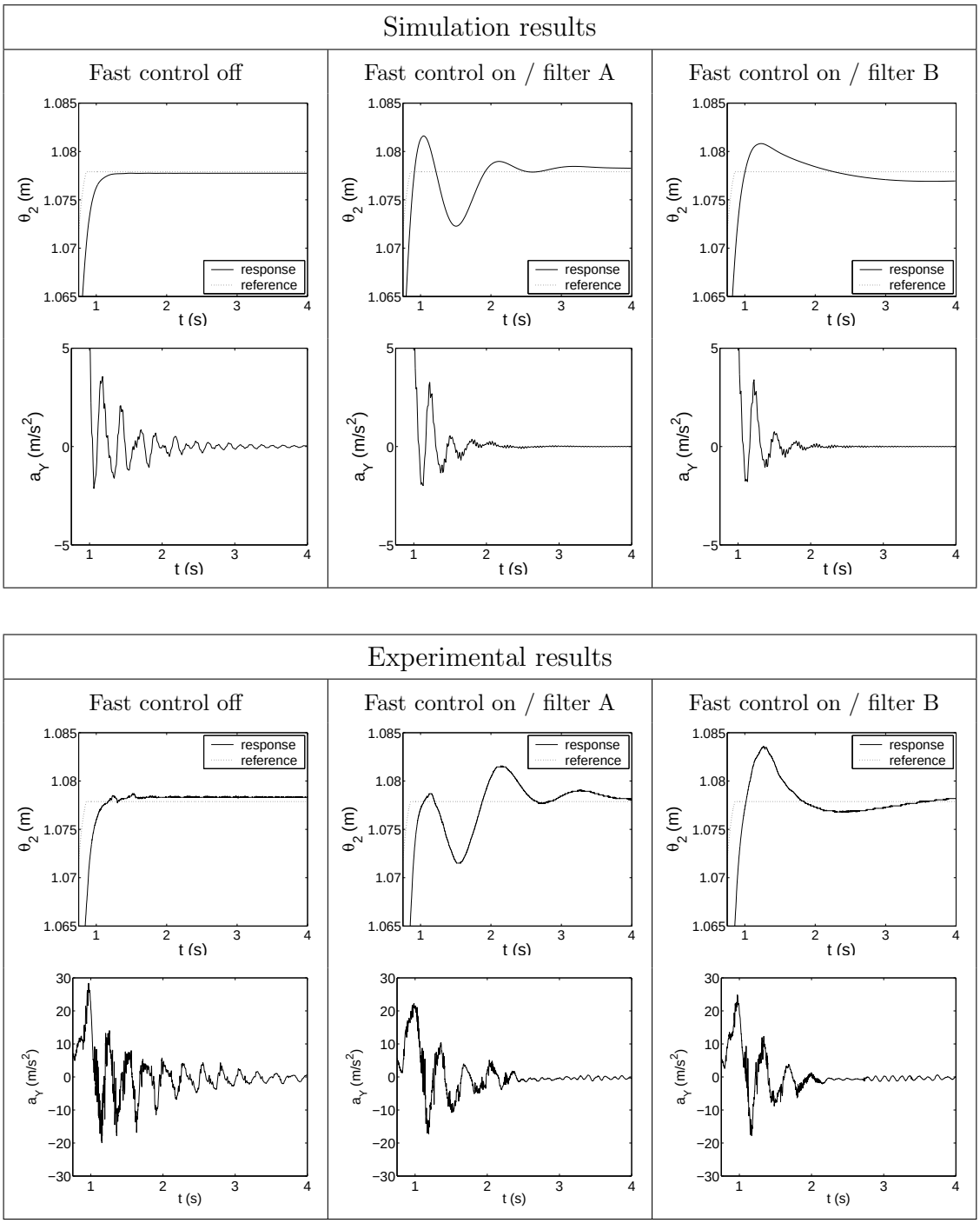


Figure 6.33: Point-to-point trajectory: zoom at the end of the trajectory, after reaching the end point.

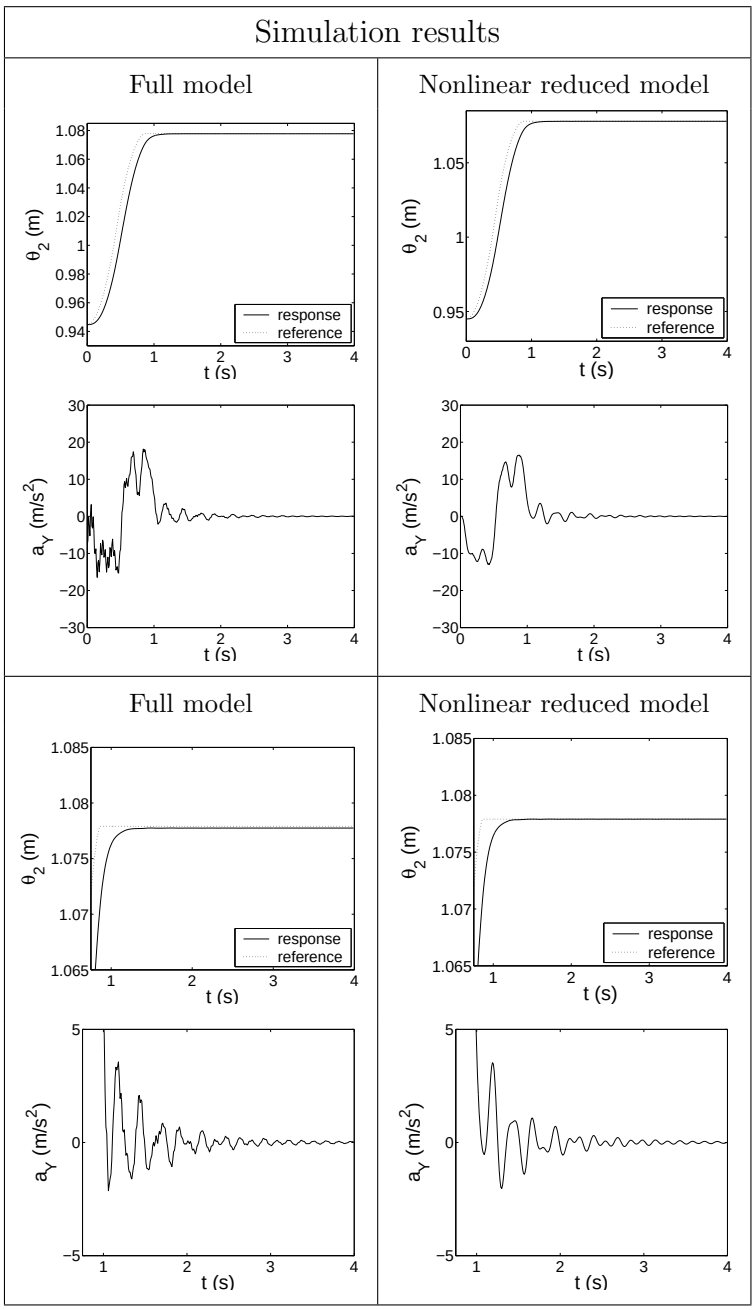


Figure 6.34: Comparison of the reduced model and the full model. Point-to-point trajectory, normal bandwidth of the slow controller, fast controller off.

relation $\dot{\mathbf{x}}^e = \boldsymbol{\Psi}^{e\boldsymbol{\eta}}\,\dot{\boldsymbol{\eta}}$, which is implemented in a special `ModesToDofs` element. In this element, the dependence of the 2×4 mode-shape matrix $\boldsymbol{\Psi}^{e\boldsymbol{\eta}}$ with respect to the configuration $\boldsymbol{\theta}$ is represented by a polynomial approximation obtained from the model reduction procedure.

The results are compared with the full model in Figure 6.34. Discrepancies mostly concern the high-frequency content of the response, but the correspondence is good for the slow component of the motion. We conclude that the model reduction technique can be utilized to speed up the simulation of a (controlled) mechanism.

Square trajectory

For a point-to-point trajectory, we are especially interested in the performances at the final configuration. Since the performances of the composite controller might depend on the configuration, a square trajectory is analyzed in the configuration space. The cyclic sequence S_1, S_2, S_3, S_4, S_1 was illustrated in Figure 6.30. The experimental results are presented in Figures 6.35 and 6.36, and the performances of the vibration controller are appreciable for each configuration.

6.5 Concluding remarks

In this chapter, the theoretical concepts developed previously have been applied to a large and lightweight manipulator, with a parallel actuation mechanism. The dynamic modeling of the mechanism, of the actuators and of the controller has been realized, validated and exploited at the different stages of a control design procedure. In summary, different models have been constructed:

- a low-order linearized model of the actuated mechanism, for experimental validation,

- a low-order and nonlinear model of the mechanism, for the design of the vibration controller, and for efficient simulation,

- a linearized model of the controlled mechanism, for root locus and frequency response analyses,

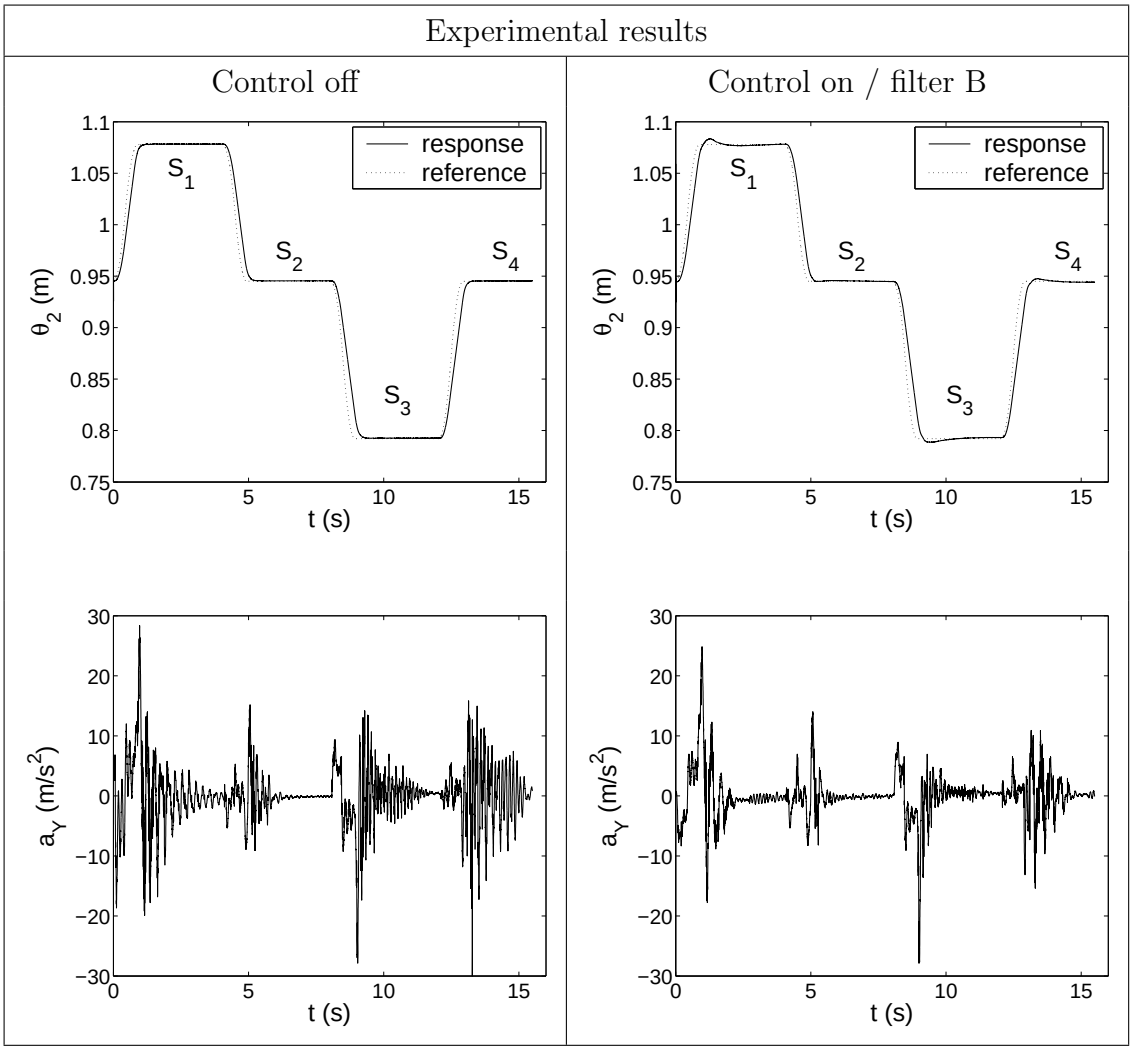


Figure 6.35: Square trajectory.

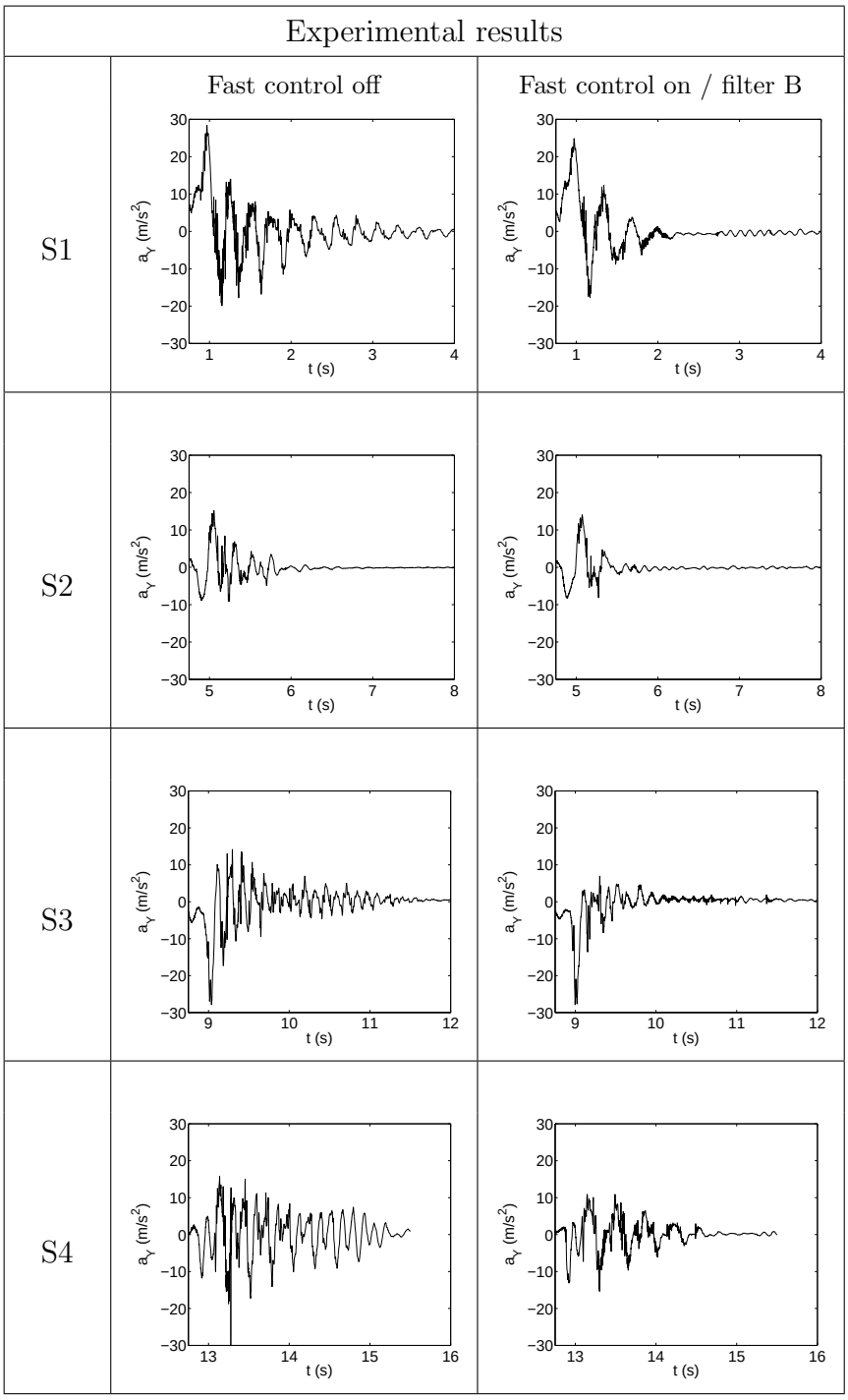


Figure 6.36: Square trajectory: various zooms.

- a full model of the mechatronic system, for the simulation of the closed-loop dynamics.

The predictions of those models were generally in good agreement with the experimental results. However, the dynamic parameters of the actuators are critical for the quality of the model. Their estimation was based on experimental data, but a more systematic identification procedure would be helpful at this level. Some discrepancies are also attributed to friction and clearance effects in the joints of the mechanism.

A two-time-scale strategy has been selected for the design of the controller. A slow controller is responsible for the tracking performances, and a fast vibration controller damps the flexible motion. The vibration controller is based on the inertial damping concept, and we conclude that:

- the inertial damping concept, initially developed for macro/micro manipulators is also applicable for any flexible manipulator,

- the implementation of this control strategy is fairly simple: only two accelerometers complement the hardware of the joint-tracking controller,

- a low-order model is necessary for the design and the implementation of the control law, our reduced-order modeling technique is appropriate for this purpose,

- one difficulty is associated with the reliable estimation of the rates of the flexible coordinates, especially at low frequencies,

- in order to prevent interferences between both controllers, filtering is necessary,

- finally, a compromise can be obtained between the efficiency of the fast vibration controller, and the performances of the slow tracking controller,

The modeling and design concepts were intimately connected throughout the chapter. The major conclusion is that *the modeling and simulation tools developed in this dissertation are highly relevant for the control design of a complex experimental mechanism.*

In future research, significant improvements of the controller could be obtained using a more accurate estimation formula, or a dynamic observer. Replacing the accelerometers by strain gauges would also lead to a more reliable signal at low frequencies, less sensitive to the rigid motion.

VII

Conclusion

The integrated design of mechatronic systems is a multidisciplinary challenge. In particular, our objective was to develop modeling and simulation methods to assist the engineer at the different stages of the design of controlled flexible mechanisms. The dissertation investigated two main research orientations: the simulation of mechatronic systems, and the reduced-order modeling of flexible mechanisms. Those concepts were also exploited for the control design of an experimental manipulator.

The specific conclusions associated with the different parts of this work are reported hereafter.

1. Integrated simulation of mechatronic systems

A modular formulation has been presented for the dynamic analysis of *mechatronic systems* composed of a flexible mechanism and a control system. The coupled differential and algebraic equations are constructed numerically, and their time-integration is performed according to a strongly coupled approach.

The methodology relies on several **strategic choices** and **personal contributions**:

- The selection of the *Finite Element method* as a modular modeling framework for the analysis of mechatronic systems, with the advantage of an accurate representation of the mechanical subsystem.

- The use of the *block diagram language* for the description of control systems in the Finite Element environment. This leads to a general, systematic and

modular formulation, familiar to control engineers.

- The implementation of the *generalized-α method* for the simulation of the coupled system. The ability of this method to deal with constrained mechanical equations or with first-order equations is well-known, and the possibility to adjust the high-frequency numerical damping is very convenient for stiff problems.

- The analysis of *stability and convergence* properties of the generalized-α method when applied to the coupled dynamic equations.

- The extension of the time-integration algorithm to deal with *discontinuous dynamic effects* in some subsystems, for instance due to sampling or saturation phenomena.

Well-established formalisms from flexible multibody dynamics and from system theory have thus been unified for the reliable and modular simulation of controlled flexible mechanisms. The most important **conclusions** are:

- The theoretical convergence and stability results are confirmed by several test-cases.

- The proposed methodology is able to predict the dynamic behavior of complex mechatronic systems, *e.g.* a car equipped with a complex semi-active suspension system, and a controlled flexible manipulator. For this last system, the simulation results are validated experimentally.

- The simulation method is a relevant pre-prototyping tool, useful for control design.

Future investigations could focus on the following issues:

- The analysis and the optimization of the computational efficiency, in particular the development of an adaptive time-stepping method.

- The combination of the approach with a multi-rate or a subcycling strategy, in order to improve the computational efficiency when very fast dynamic phenomena have to be represented within specific subsystems.

- The extension of the functional simulation concept to include subsystem models developed in formalisms dedicated to specific technologies (*e.g.* electrical, hydraulic or pneumatic systems), which would reinforce the modularity and the generality of the methodology.

- The development of a user-friendly graphical interface, and of drivers with software specialized in block diagram modeling.

2. Model reduction in flexible multibody dynamics

In order to represent concisely the configuration-dependent dynamics of a *flexible mechanism*, a systematic model reduction technique has been proposed, which transforms an initial high-order model into a low-order and explicit model. The reduction procedure relies on (i) an order reduction, which can be realized locally for any given configuration, (ii) a *consistent* approximation of the reduced-order model in the configuration space.

Since similar approaches are reported in the literature, we need to draw the attention to the **originality** of our methodology:

- As the *Finite Element method* is selected to formulate the dynamics of the initial model, complex mechanical topologies can be systematically represented.

- The *Global Modal Parameterization* concept is formally defined in order to guarantee the consistency of the reduction procedure and the physical interpretation of the reduced coordinates. This parameterization is based on an extension of the component-mode method, and it accounts for the nonlinear kinematics. The Global Modal Parameterization is well-defined in the configuration space of interest, and it represents exactly the rigid-body motion.

- The local reduction procedure accounts for the *nonlinearity* of the Global Modal Parameterization; in particular, the curvature of the parameterization contributes to the equivalent inertia forces. Nonlinear effects caused by large deformations are neglected.

- A *piecewise polynomial approximation* of the reduced-order model in the configuration space follows an adaptive strategy, which achieves a user-specified tolerance on the approximation error.

A complex flexible mechanism can thus be represented by a compact and closed-form dynamic model, formulated in a systematic way. From the development of the methodology and its application to various problems, **we conclude that**:

- The model leads to a deep understanding of the dynamic system under consideration. Valuable insights are obtained into the inertia, stiffness and kinematic properties all over the configuration space.

- The degree of nonlinearity of the model is associated with the degree of nonlinearity of the dynamics seen by the actuators; the reduction method breaks down at the actuator singularities.

- Low-order polynomials are able to capture efficiently the smooth variations of the model in the configuration space. A compromise is necessary between accuracy, continuity, memory storage requirements and computational time to build the model. Lagrange polynomials defined on a regular grid have ideal continuity properties, but this strategy requires important resources to capture a nonlinear behavior in a high-dimensional configuration space. For such specific problems, adaptive configuration space decomposition and/or quadratic polynomials may be preferred.

- The method is applicable for rigid mechanisms with parallel topology; the reduced model gives the solution to the inverse dynamics problem, and it is directly exploitable for computed-torque control techniques.

- The configuration-dependent dynamics of a flexible manipulator is efficiently captured by a reduced model, as confirmed by experimental results.

- A reduced model is an appropriate basis for the design of a vibration controller, *e.g.* using the inertial damping concept. It is portable and it can be implemented online on a real-time controller.

- The computational efficiency of a simulation algorithm is improved using a reduced model.

Several **outcomes** are foreseen for the model reduction technique:

- The reduction method can be a basis for the extraction of a more structured model (*e.g.*, a polytopic linear model or a model based on linear fractional transformation), for which specialized control theories are applicable.

- In the literature related with component-mode synthesis, various enhanced mode shapes have been defined; any of them could be associated with the Global Modal Parameterization.

- A priori knowledge on the configuration space geometry (specified trajectory, symmetries, etc) could be exploited for an optimized approximation of the nonlinear model.

- As such, our formalism is applicable to mechanisms without pre-stressing effects. The extension to include those effects could also be considered in future investigations.

3. Control of a flexible manipulator

The proposed simulation and modeling methods have been applied for the motion and vibration control of a flexible manipulator.

The design of the controller is based on the following **personal developments**:

- A *two-time-scale control strategy* is elaborated from the structure of the dynamic model: the rigid modes are controlled by a slow joint-tracking controller, and the flexible modes by a fast vibration controller.

- The *inertial damping concept* is applied for the vibration control of the flexible manipulator.

- The *interaction problem* caused by a non-perfect time-scale separation is analyzed, and solved by appropriate filtering.

Various models are involved in the control design procedure, which confirms the relevance of our original modeling and simulation tools. From the simulation and experimental results, **we conclude that**:

- The identification of the parameters of the mechanical model is important to obtain a good agreement with the experimental data. This motivates the future development of systematic identification techniques in flexible multibody dynamics, which can take advantage of the model reduction method.

- Inertial damping leads to a simple, pragmatic and efficient control strategy, able to improve the dynamic behavior of a mechanism proned to vibration problems.

- Due to the necessary time-scale separation, a compromise is necessary between motion performances and vibration suppression.

- The reliable estimation of the modal velocities is critical for the performance of the controller. Improvements are expected if the acceleration feedback is replaced by a strain feedback, less sensitive to low-frequency disturbances.

4. General perspective and closure

The modeling and simulation tools presented in this dissertation appear as a relevant basis for constructive methods in mechatronics. In particular, trajectory generation, inverse dynamics and concurrent design are challenging problems for controlled mechanisms, for which optimization techniques can be of great assistance. The utilization of our methods in a numerical optimization approach is thus a recommended direction for future research.

Mechatronics is a multidisciplinary framework; at the closure of this thesis, we strongly believe that our contribution naturally resulted from the cross-fertilization of concepts encountered in several fields of science and technology. More than ever, discarding inter-domain barriers appears as a fruitful guideline in engineering.

Acknowledgements

This work has been supported by the Belgian National Fund for Scientific Research (FNRS) which is gratefully acknowledged. It also presents research results of the Belgian Program on Inter-University Poles of Attraction initiated by the Belgian state, Prime Minister's office, Science Policy Programming (AMS IAP V/06). The scientific responsibility is assumed by its author.

Bibliography

[Ang01] G.Z Angelis. *System Analysis, Modelling and Control with Polytopic Linear Models*. PhD thesis, Technische Universiteit Eindhoven, 2001.

[Arn01] M. Arnold. Numerically stable modular time integration of multiphysical systems. In K.J. Bathe, editor, *Proceedings of the First MIT Conference on Computational Fluid and Solid Mechanics*, pages 1062–1064, Cambridge, MA, June 2001. Elsevier, Amsterdam.

[Arn04] M. Arnold. Numerical methods in the simulation of vehicle-guideway interaction. In A. Buikis, R. Ciegis, and A.D. Fitt, editors, *Progress in Industrial Mathematics at ECMI 2002*, pages 115–120, Berlin Heidelberg, 2004. Springer–Verlag.

[AS86] H. Asada and J.-J.E. Slotine. *Robot Analysis and Control*. Wiley-Interscience, 1986.

[Bau72] J. Baumgarte. Stabilization of constraints and integrals of motion in dynamical systems. *Comp. Meth. Appl. Mech. Eng.*, 1:1–16, 1972.

[Bay87] E. Bayo. A finite element approach to control the end-point motion of a single-link flexible robot. *Int. J. Robotic Systems*, 4(1):63–75, 1987.

[BBT03] O.A. Bauchau, C.L. Bottasso, and L. Trainelli. Robust integration schemes for flexible multibody systems. *Comp. Meth. Appl. Mech. Eng.*, 192:395–420, 2003.

[BC04] C.L. Bottasso and A. Croce. Optimal control of multibody systems using an energy preserving direct transcription method. *Multibody System Dynamics*, 12:17–45, 2004.

[BCGF04] C.L. Bottasso, A. Croce, L. Ghezzi, and P. Faure. On the solution of inverse dynamics and trajectory optimization problems for multibody systems. *Multibody System Dynamics*, 11:1–22, 2004.

[BCP96] K.E. Brenan, S.L. Campbell, and L.R. Petzold. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. SIAM, Philadelphia, 2nd edition, 1996.

[BDG02] O. Brüls, P. Duysinx, and J.-C. Golinval. An adaptation of the Newmark scheme for the integrated simulation of mechatronic systems. In *Proc. of the 6th Int.*

Conf. on Motion and Vibration Control (MOVIC), volume 1, pages 484–489, Saitama, Japan, August 2002.

[BDG03] O. Brüls, P. Duysinx, and J.-C. Golinval. Computational environment for the design of flexible mechanisms with feedback control. In *Proc. of the 6th National Congress on Theoretical and Applied Mechanics*, Ghent, Belgium, May 2003.

[BDG04a] O. Brüls, P. Duysinx, and J.-C. Golinval. Generation of closed-form models for the control of flexible mechanisms: a numerical approach. In *Proc. of the 7th Int. Conf. on Motion and Vibration Control (MOVIC)*, Saint-Louis, United States, August 2004.

[BDG04b] O. Brüls, P. Duysinx, and J.-C. Golinval. Reliable simulation of mechatronic systems using newmark algorithms. In *Colloquium of the European Mechanics Society on Advances in Simulation Techniques for Applied Dynamics (EUROMECH 452) - Book of Abstracts*, Halle, Germany, March 2004.

[BDG04c] O. Brüls, P. Duysinx, and J.-C. Golinval. A systematic model reduction method for the control of flexible multibody systems. In *Proc. of the 21st Int. Congress of Theoretical and Applied Mechanics (ICTAM)*, Warsaw, Poland, August 2004.

[BDG06] O. Brüls, P. Duysinx, and J.-C. Golinval. A model reduction method for the control of rigid mechanisms. *Multibody System Dynamics*, in press:–, 2006.

[BG03] O. Brüls and J.-C. Golinval. Simulation of an active control system in a hot-dip galvanizing line. In W. Schiehlen and M. Valásek, editors, *Virtual Nonlinear Multibody Systems*, volume 103 of *NATO Science Series II: Mathematics, Physics and Chemistry*, pages 325–332. Kluwer Academic Publishers, 2003.

[Boo84] W.J. Book. Recursive Lagrangian dynamics of flexible manipulator arms. *Int. J. Robotics Res.*, 3(3):87–101, 1984.

[Boo93] W.J. Book. Controlled motion in an elastic world. *ASME J. Dyn. Syst., Meas., and Control*, 115(2B):252–261, 1993.

[BP87] E. Bayo and B. Paden. On trajectory generation for flexible robots. *Int. J. Robotic Systems*, 4(2):229–235, 1987.

[Car89] A. Cardona. *An Integrated Approach to Mechanism Analysis*. PhD thesis, Université de Liège, 1989.

[CB68] R. Craig and M. Bampton. Coupling of substructures for dynamic analysis. *AIAA J.*, 6(7):1313–1319, 1968.

[CG88] A. Cardona and M. Géradin. A beam finite element non-linear theory with finite rotations. *Int. J. Num. Meth. Eng.*, 26:2403–2438, 1988.

[CG89] A. Cardona and M. Géradin. Time integration of the equations of motion in mechanism analysis. *Computers and Structures*, 33:801–820, 1989.

[CG91] A. Cardona and M. Géradin. A superelement formulation for mechanism analysis. *Int. J. Num. Meth. Eng.*, 32(8):1565–1594, 1991.

[CH93] J. Chung and G.M. Hulbert. A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-α method. *ASME J. Appl. Mech.*, 60:371–375, 1993.

[CKG94] A. Cardona, I. Klapka, and M. Géradin. Design of a new finite element programming environment. *Engineering Computations*, 11:365–381, 1994.

[Cra87] R. Craig. A review of time-domain and frequency domain component-mode synthesis methods. *Int. J. Analytical and Experimental Modal Analysis*, 11(6):562–570, 1987.

[CS84] H.R. Cannon and E. Schmitz. Initial experiments on the end-point control of a flexible one-link robot. *Int. J. Robotics Res.*, 3(3):62–75, 1984.

[CU02] F.M. Caswara and H. Unbehauen. A neurofuzzy approach to the control of a flexible-link manipulator. *IEEE Trans. Robotics and Automation*, 18(6):932–944, December 2002.

[CW03] D. Chablat and Ph. Wenger. Architecture optimization of a 3-dof translational parallel mechanism for machining applications, the Orthoglide. *IEEE Trans. Robotics and Automation*, 19(3):403– 410, June 2003.

[CWA00] D. Chablat, Ph. Wenger, and J. Angeles. Conception isotropique d'une morphologie parallèle : Application à l'usinage. In *Proc. of the 3rd Int. Conf. On Integrated Design and Manufacturing in Mechanical Engineering*, Montreal, Canada, May 2000.

[CWM02] D. Chablat, Ph. Wenger, and J-P Merlet. Workspace analysis of the orthoglide using interval analysis. In *Proc. of the 8th Int. Symp. on Advances in Robot Kinematics*, Caldes de Malavella, Spain, June 2002. Kluwer Academic Publishers.

[CYC02] J. Cheong, Y. Youm, and W.K. Chung. Joint tracking controller for multi-link flexible robot using disturbance observer and parameter adaptation scheme. *Int. J. Robotic Systems*, 19(8):401–417, 2002.

[DB94] S. Devasia and E. Bayo. Inverse dynamics of articulated flexible structures : Simultaneous trajectory tracking and vibration reduction. *J. Dynamics and Control*, 4(3):299–309, 1994.

[DF00] P. De Fonseca. *Simulation and Optimisation of the Dynamic Behaviour of Mechatronic Systems.* PhD thesis, Katholieke Universiteit Leuven, 2000.

[DJB94] J.G. De Jalón and E. Bayo. *Kinematic and Dynamic Simulation of Multibody Systems - The Real-Time Challenge.* Springer Verlag, 1994.

[DLS93] A. De Luca and B. Siciliano. Inverse-based nonlinear control of robot arms with flexible links. *AIAA J. Guidance, Control, and Dynamics*, 16(6):1169–1176, 1993.

[DV76] B.F. De Veubeke. The dynamics of flexible bodies. *Int. J. Engineering Science*, 14:895–913, 1976.

[EBB02] S. Erlicher, L. Bonaventura, and O.S. Bursi. The analysis of the generalized-α method for non-linear dynamic problems. *Computational Mechanics*, 28:83–104, 2002.

[Eck99] H. Ecker. Comparison 11 (scara-robot) with acsl, full hybrid modelling approach - environment level. *Simulation News Europe*, 26:43, July 1999.

[EMO98] H. Elmqvist, S.E. Mattson, and M. Otter. Modelica - the new object-oriented modeling language. In *Proc. of the 12th European Simulation Multiconference (ESM'98)*, June 1998.

[ES98] P. Eberhard and W. Schiehlen. Hierarchical modeling in multibody dynamics. *Arch. of Appl. Mech.*, 68:237–246, 1998.

[Fav97] W. Favre. *Contribution à la Représentation Bond Graph des Systèmes Mécaniques Multicorps*. PhD thesis, INSA de Lyon, 1997.

[FLMR99] M. Fliess, J. Lévine, P. Martin, and P. Rouchon. A Lie-Bäcklund approach to equivalence and flatness of nonlinear systems. *IEEE Trans. Autom. Control*, 44(5):922–937, May 1999.

[FLOR95] M. Fliess, J. Lévine, F. Ollivier, and P. Rouchon. Flatness and dynamic feedback linearizability: Two approaches. In *Proc. 3rd European Control Conference*, 1995.

[For98] E. Forsthuber. Comparison 11 (scara-robot) with dymola, classical mechanical approach, automatic-symbolical and numerical inversion. *Simulation News Europe*, 23:49, July 1998.

[Gaw98] W. K. Gawronski. *Dynamics and Control of Structures - A Modal Approach*. Springer, New York, 1998.

[GC01] M. Géradin and A. Cardona. *Flexible Multibody Dynamics: A Finite Element Approach*. John Wiley & Sons, New York, 2001.

[Gea71] C.W. Gear. *Numerical Initial Value Problems in Ordinary Differential Equations*. Prentice-Hall, New Jersey, 1971.

[Geo02] L.E. George. *Active Vibration Control of a Flexible Base Manipulator*. PhD thesis, Georgia Institute of Technology, 2002.

[GKCW02] S. Guegan, W. Khalil, D. Chablat, and P. Wenger. Modélisation dynamique d'un robot parallèle à 3-ddl : l'Orthoglide. In *Conférence Internationale Francophone d'Automatique*, Nantes, July 2002.

[GLL98] L.B. Gutiérrez, F.L. Lewis, and J.A. Lowe. Implementation of a neural network tracking controller for a single flexible link: Comparison with PD and PID controllers. *IEEE Trans. Industrial Electronics*, 45(2):307–318, april 1998.

[GLZ96] S.S. Ge, T.H. Lee, and G. Zhu. Energy-based robust controller design for multi-link flexible robots. *Mechatronics*, 6(7):779–798, 1996.

[GR97] M. Géradin and D. Rixen. *Mechanical Vibrations: Theory and Application to Structural Dynamics*. John Wiley & Sons, New York, 2nd edition, 1997.

[GS00] F. Ghorbel and M.W. Spong. Integral manifolds of singularly perturbed systems with application to rigid-link flexible-joint multibody systems. *Int. J. of Non-Linear Mechanics*, 35:133–155, 2000.

[GVVD04] K. Gallivan, A. Vandendorpe, and P. Van Dooren. Model reduction of MIMO systems via tangential interpolation. *SIAM J. Matrix Analysis and Applications*, 26(2):328–349, 2004.

[GW84] C.W. Gear and D.R. Wells. Multirate linear multistep methods. *BIT*, 24:482–502, 1984.

[HAV03] A. Heckmann, M. Arnold, and O. Vaculin. Distributed multiphysical phenomena in multibody dynamics. In *Proc. of the ECCOMAS Conf. on Advances in Computational Multibody Systems*, Lisbon, Portugal, July 2003.

[Her79] D.N. Herting. A general purpose, multi-stage, component-mode synthesis method. In *Proceedings of the AIAA/ASME/ASCE/AHS 20th Structure, Structural Dynamics and Material Conference*, 1979.

[HHT77] H. Hilber, T. Hughes, and R. Taylor. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering and Structural Dynamics*, 5:283–292, 1977.

[HNW87] E. Hairer, S.P. Norsett, and G. Wanner. *Solving Ordinary Differential Equations I - Nonstiff Problems*. Springer-Verlag, 1987.

[Hol80] J.M. Hollerbach. A recursive formulation of lagrangian manipulator dynamics. *IEEE Trans. Syst., Man, Cybern.*, 10(11):730–736, 1980.

[HP95] V. Hadwitch and K. Pfeiffer. The principle of virtual work in mechanical and electromechanical systems. *Arch. of Appl. Mech.*, 65:390–400, 1995.

[Hug87] T.J.R. Hughes. *The Finite Element Method, Linear Static and Dynamics Finite Element Analysis*. Prentice Hall, 1987.

[Hug88] J.D. Huggins. Experimental verification of a model of a two-link flexible, lightweight manipulator. Master's thesis, Georgia Institute of Technology, 1988.

[Hur65] W.C. Hurty. Dynamic analysis of structural systems using component modes. *AIAA J.*, 3(4):678–685, 1965.

[Hus91] R.L. Huston. Computer methods in flexible multibody dynamics. *Int. J. Num. Meth. Eng.*, 32:1657–1668, 1991.

[HW91] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II - Stiff and Differential-Algebraic Problems*. Springer-Verlag, 1991.

[Isi95] Isidori. *Nonlinear Control System*. Springer, 1995.

[JKDW01] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter. *Applied Interval Analysis*. Springer Verlag, London, 2001.

[JWH00] K.E. Jansen, C.H. Whiting, and G.M. Hulbert. A generalized-a method for integrating the filtered navier-stokes equations with a stabilized finite element method. *Comp. Meth. Appl. Mech. Eng.*, 190:305–319, 2000.

[Kar97] D. Karnopp. Understanding multibody dynamics using bond graph representations. *J. Franklin Inst.*, 334B(4):631–642, 1997.

[KB94] D.S. Kwon and W.J. Book. A time-domain inverse dynamic tracking control of a single-link flexible manipulator. *ASME J. Dyn. Syst., Meas., and Control*, 116(2):193–200, 1994.

[KJT95] F. Khorrami, S. Jain, and A. Tzes. Experimental results on adaptive nonlinear control and input preshaping for multi-link flexible manipulators. *Automatica*, 31(3):83–97, 1995.

[KK00] T.S. Kim and Y.Y. Kim. MAC-based mode-tracking in structural topology optimization. *Computers and Structures*, 74:375–383, 2000.

[KMR00] D. Karnopp, D.L. Margolis, and R.C. Rosenberg. *System Dynamics: Modeling and Simulation of Mechatronic Systems*. John Wiley & Sons Inc., 3rd edition, 2000.

[KS00] R. Kübler and W. Schiehlen. Two methods of simulator coupling. *Mathematical and Computer Modelling of Dynamical Systems*, 6(2):93–113, 2000.

[KTKH67] H. E. Koenig, Y. Tokad, H. K. Kesavan, and H. G. Hedges. *Analysis of Discrete Physical Systems*. McGraw-Hill Book Company, New York NY, 1967.

[Lan79] Y.D. Landau. *Adaptive Control - The Model Reference Approach*. Marcel Dekker, 1979.

[LCG04] E.V. Lens, A. Cardona, and M. Géradin. Energy preserving time integration for constrained multibody systems. *Multibody System Dynamics*, 11(1):41–61, 2004.

[Lee90] J.W. Lee. *Dynamic Analysis and Control of Lightweight Manipulators With Flexible Parallel Link Mechanism*. PhD thesis, Georgia Institute of Technology, 1990.

[LHKB91] H.-J. Lai, E.J. Haug, S.S. Kim, and D.-S. Bae. A decoupled flexible-relative coordinate recursive approach for flexible multibody dynamics. *Int. J. Num. Meth. Eng.*, 32:1669–1690, 1991.

[LLL03] G. Liu, Y. Lou, and Z. Li. Singularities of parallel manipulators: A geometric treatment. *IEEE Trans. Robotics and Automation*, 19(4):579–594, 2003.

[LSS04] C. Lauwerys, J. Swevers, and P. Sas. Model free design for a semi-active suspension of a passenger car. In *Proc. of ISMA 2004*, Leuven, Belgium, September 2004.

[LWP80] J.Y.S. Luh, M.W. Walker, and R.P.C. Paul. On-line computational scheme for mechanical manipulators. *ASME J. Dyn. Syst., Meas., and Control*, 102:69–76, 1980.

[Mag91] D.P. Magee. Dynamic control in teleoperation of a flexible manipulator. Master's thesis, Georgia Institute of Technology, 1991.

[MC95] L. Meirovitch and Y. Chen. Trajectory and control optimization for flexible space robots. *AIAA J. Guidance, Control, and Dynamics*, 18(3):493–502, 1995.

[McP96] J.J. McPhee. On the use of linear graph theory in multibody system dynamics. *Nonlinear Dynamics*, 9:73–90, 1996.

[McP98] J.J. McPhee. Automatic generation of motion equations for planar mechanical systems using the new set of "branch coordinates". *Mech. Mach. Theory*, 33(6):805–823, 1998.

[McP03] J.J McPhee. Virtual prototyping of multibody systems with linear graph theory and symbolic computing. In W. Schiehlen and M. Valásek, editors, *Virtual Nonlinear Multibody Systems*, volume 103 of *NATO Science Series II : Mathematics, Physics and Chemistry*, pages 37–56. Kluwer Academic Publishers, 2003.

[MEFK97] P. Maisser, O. Enge, G. Freundenberg, and G. Kielau. Electromechanical interactions in multibody systems containing electromechanical drives. *Multibody System Dynamics*, 1(3):281–302, 1997.

[Mei90] L. Meirovitch. *Dynamics and Control of Structures.* John Wiley & Sons, New-York, 1990.

[MK91] L. Meirovitch and M.-K. Kwak. Rayleigh-ritz based substructure synthesis for flexible multibody dynamics. *AIAA J.*, 28:1709–1719, 1991.

[MM95] R.H. Myers and D.C. Montgomery. *Response Surface Methodology.* Wiley Series in Probability and Statistics, 1995.

[Mon97] D.C. Montgomery. *Design and Analysis of Experiments.* John Wiley & Sons, New York, 4th edition, 1997.

[MPK00] M. Moallem, R.V. Patel, and K. Khorasani. *Flexible-link Robot Manipulators - Control Techniques and Structural Design.* Springer, 2000.

[MS94] R. Murray-Smith. *A Local Model Network Approach to Nonlinear Modelling.* PhD thesis, University of Strathclyde, Department of Computer Science, Glasgow, Scotland, November 1994.

[Nel99] O. Nelles. *Nonlinear System Identification with Local Linear Neuro-Fuzzy Models.* PhD thesis, TU Darmstadt, 1999.

[New59] N. Newmark. A method of computation for structural dynamics. *ASCE J. Eng. Mech. Div.*, 85:67–94, 1959.

[Nik03] P. Nikravesh. The use of principal axes as floating reference frame for a moving deformable body. In *Proc. of the ECCOMAS Conf. on Advances in Computational Multibody Systems*, Lisbon, Portugal, July 2003.

[Obe98] Obergfell. End-point position sensing and control of flexible multi-link manipulators. Master's thesis, Georgia Institute of Technology, 1998.

[OC89] C.M. Oakley and R.H. Cannon. End-point control of a two-link manipulator with a very flexible forearm: Issues and experiments. In *Proc. of the 1989 American Control Conference*, volume 2, pages 1381–1388, Pittsburgh, PA, USA, June 1989.

[OE97] M. Otter and H. Elmqvist. Energy flow modeling of mechatronic systems via object diagrams. In *Proc. of the 2nd Mathmod Vienna, IMACS Symposium on Mathematical Modelling*, pages 705–710, 1997.

[OL99] J.H. Oh and J.S. Lee. Control of flexible joint robot system by backstepping design approach. *Automation and Soft Computing*, 5(4):267–278, 1999.

[OS95] B. Owren and H.H. Simonsen. Alternative integration methods for problems in structural dynamics. *Comp. Meth. Appl. Mech. Eng.*, 122:1–10, 1995.

[OS01] F. Ollivier and A. Sedoglavic. A generalization of flatness to nonlinear systems of partial differential equations. application to the command of a flexible rod. In *Proceedings of the 5th IFAC Symposium on Nonlinear Control Systems*, volume 1, pages 196–200, Saint Petersburg, Russia, July 2001. Elsevier.

[PK99] F.C. Park and J.W. Kim. Singularity analysis of closed kinematic chains. *ASME J. Mech. Des.*, 121(1):32–38, 1999.

[PLS03] J.R. Phillips, D. Luca, and L.M. Silveira. Guaranteed passive balancing transformations for model order reduction. *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, 22(8):1027– 1041, August 2003.

[Pre97] A. Preumont. *Vibration Control of Active Structures, An Introduction.* Kluwer, 1997.

[PTVF92] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C - The Art of Scientific Computing.* Cambridge University Press, 2nd edition, 1992.

[Rai77] M.H. Raibert. Analytical equations vs. table look-up for manipulation: a unifying concept. In *Proc. of the IEEE Conf. Decision and Control*, New Orleans, 1977.

[RS98] A. Rükgauer and W. Schiehlen. Simulation of modular dynamic systems. *Computers and Structures*, 46:535–542, 1998.

[RV02] I. Romero and R. Vignjevic. Deployable space manipulator closed loop control, ideas and possibilities of using gps as sensor. *Advances in Space Research*, 30(2):419–425, 2002.

[SAM99] SAMTECH, Liège, Belgium. *SAMCEF Users Manual - v 8.0*, 1999.

[Sas04] L. Sass. *Symbolic Modeling of Electromechanical Multibody Systems*. PhD thesis, Université Catholique de Louvain, 2004.

[SB88] B. Siciliano and W.J. Book. A singular perturbation approach to control of lightweight fexible manipulators. *Int. J. Robotics Res.*, 7(4):79–90, 1988.

[Sch97] W. Schiehlen. Multibody system dynamics: Roots and perspectives. *Multibody System Dynamics*, 1:149–188, 1997.

[Sch98] J. Scheikl. Comparison 11 (scara-robot) with matlab/simulink, hybrid modelling approach, model level. *Simulation News Europe*, 24:44, November 1998.

[Sch01] C.W. Scherer. LPV control and full block multipliers. *Automatica*, 37:361–375, 2001.

[Sco95] M.A. Scott. *Time Varying Compensator Design for Reconfigurable Structures using Non-Collocated Feedback*. PhD thesis, University of Colorado, 1995.

[SF03] J.-C. Samin and P. Fisette. *Symbolic Modeling of Multibody Systems*. Kluwer Academic Publishers, 2003.

[SH04] K. Seto and M. Hon. Modeling strategy suited to motion and vibration control for flexible structures. In *Proc. of the 7th Int. Conf. on Motion and Vibration Control (MOVIC)*, St-Louis, USA, August 2004.

[Sha98] A.A. Shabana. *Dynamics of Multibody Systems*. Cambridge University Press, 2nd edition, 1998.

[Sim85] J.C. Simo. A finite strain beam formulation. the three-dimensional dynamic problem. Part I. *Comp. Meth. Appl. Mech. Eng.*, 49:55–70, 1985.

[SJK97] R. Sepulchre, M. Jankovic, and P. Kokotovic. *Constructive Nonlinear Control*. Springer-Verlag Series in Communications and Control Engineering, 1997.

[SL91] J.-J.E. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, 1991.

[SLP03] R. Serban, D. Li, and L.R. Petzold. Adaptive algorithms for optimal control of time-dependent partial differential-algebraic equation systems. *Int. J. Num. Meth. Eng.*, 57(2):1457–1469, 2003.

[SM00] P. Shi and J. McPhee. Dynamics of flexible multibody systems using virtual work and linear graph theory. *Multibody System Dynamics*, 4:355–381, 2000.

[SM03a] M. Scherrer and J. McPhee. Dynamic modelling of electromechanical multibody systems. *Multibody System Dynamics*, 9:87–115, 2003.

[SM03b] C. Schmitke and J. McPhee. Modelling mechatronic multibody systems using symbolic subsystems models. In *Proc. of the ECCOMAS Conf. on Advances in Computational Multibody Systems*, Lisbon, Portugal, July 2003.

[SMH01] P. Shi, J. McPhee, and G. Heppler. A deformation field for Euler-Bernoulli beams with applications to flexible multibody dynamics. *Multibody System Dynamics*, 5:79–104, 2001.

[SP01] R. Serban and L.R. Petzold. COOPT - a software package for optimal control of large-scale differential-algebraic equation systems. *Mathematics and Computers in Simulation*, 56(2):187–203, 2001.

[SS90] N.C. Singer and W.P. Seering. Preshaping command inputs to reduce system vibration. *ASME J. Dyn. Syst., Meas., and Control*, 112:76–82, 1990.

[Str98] O. von Stryk. Optimal control of multibody systems in minimal coordinates. *Z. Angew. Math. Mech*, 78:1117–1120, 1998.

[STW92] J.C. Simo, N. Tarnow, and K.K. Wong. Exact energy-momentum conserving algorithms and symplectic schemes for nonlinear dynamics. *Comp. Meth. Appl. Mech. Eng.*, 100:63–116, 1992.

[SV01] B. Siciliano and L. Villani. Two-time scale force and position control of flexible manipulators. In *Proc. of the 2001 IEEE Int. Conf. on Robotics and Automation*, pages 2729–2734, Seoul, Korea, 2001.

[SVQ86] J.C. Simo and L. Vu-Quoc. A three-dimensional finite-strain rod model. Part II: Computational aspects. *Comp. Meth. Appl. Mech. Eng.*, 58:79–116, 1986.

[SW83] A.A. Shabana and R.A. Wehage. A coordinate reduction technique for dynamic analysis of spatial substructures with large angular rotations. *J. Struct. Mech.*, 11:401–431, 1983.

[Sym04] W. Symens. *Motion and Vibration Control of Mechatronic Systems with Variable Configuration and Local Non-Linear Friction*. PhD thesis, Katholieke Universiteit Leuven, 2004.

[Til01] M.M. Tiller. *Introduction to Physical Modeling with Modelica*. Kluwer Academic Publishers, 2001.

[TOB01] M. Thümmel, M. Otter, and J. Bals. Control of robots with elastic joints based on automatic generation of inverse dynamics models. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Maui, Hawaii, USA, October 2001.

[TS85] T. Takagi and M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. Syst., Man, Cybern.*, 15:116–132, 1985.

[VA99] A. Veitl and M. Arnold. Coupled simulation of multibody systems and elastic structures. In J.A.C. Ambrosio and W.O. Schiehlen, editors, *Advances in Computational Multibody Dynamics*, pages 635–644, IDMEC/IST, Lisbon, Portugal, September 1999.

[VB96] H. Van Brussel. Mechatronics - a powerful concurrent engineering framework. *IEEE/ASME Trans. Mechatronics*, 1(2):127–136, 1996.

[VBS02] H. Van Brussel and W. Symens. Vibration and motion control of mechatronic systems. In *Proc. of the 2nd Int. Conf. on Advanced Computational Methods in Engineering (ACOMEN)*, Liège, Belgium, May 2002.

[VBSN+01] H. Van Brussel, P. Sas, I. Németh, P. De Fonseca, and P. Van den Braembussche. Towards a mechatronic compiler. *IEEE/ASME Trans. Mechatronics*, 6(1):90–105, 2001.

[VBSV99] M. Valásek, P. Breedveld, Z. Sika, and T. Vampola. Software tools for mechatronic vehicles: Design through modelling and simulation. *Vehicle System Dynamics*, 33:214–230, 1999.

[VGVDS+99] A. Veitl, T. Gordon, A. Van De Sand, M. Howell, M. Valásek, O. Vaculin, and P. Steinbauer. Methodologies for coupling simulation models and codes in mechatronic system analysis and design. *Vehicle System Dynamics*, 33:231–243, 1999.

[VNM98] M.J. Van Nieuwstadt and R.M. Murray. Real-time trajectory generation for differentially flat systems. *Int. J. Robust Nonlinear Control*, 18(11):995–1020, 1998.

[WC00] P. Wenger and D. Chablat. Kinematic analysis of a new parallel machine tool: the Orthoglide. In *Proc. of the 7th International Symposium on Advances in Robot Kinematics*, Slovenia, June 2000.

[WH82] R.A. Wehage and E.J. Haug. Generalized coordinate partitioning for dimension reduction in analysis of constrained dynamic systems. *ASME J. Mech. Des.*, 104:247–255, 1982.

[Wil86] T.R. Wilson. The design and construction of flexible manipulators. Master's thesis, Georgia Institute of Technology, 1986.

[WV91a] D. Wang and M. Vidyasagar. Control of a class of manipulators with a single flexible link : Part I. feedback linearization. *ASME J. Dyn. Syst., Meas., and Control*, 113:655–661, 1991.

[WV91b] D. Wang and M. Vidyasagar. Transfer functions for a single flexible link. *Int. J. Robotics Res.*, 10:540–549, 1991.

[WW95] E. Wittbrodt and S. Wojciech. Application of rigid finite element method to dynamic analysis of spatial systems. *AIAA J. Guidance, Control, and Dynamics*, 18(4):891–898, 1995.

[YBS89] B.-S. Yuan, W.J. Book, and B. Siciliano. Direct adaptive control of a one-link flexible arm with tracking. *Int. J. Robotic Systems*, 6(6):663–680, 1989.

[ZBG01] D. Zlatanov, I. Bonev, and C. Gosselin. Constraint singularities as configuration space singularities, 2001. http://www.parallemic.org/Reviews/Review008.html.

[ZT89] O.C. Zienkiewicz and R.L. Taylor. *The finite element method. Vol. I. Basic formulations and linear problems.* McGraw-Hill, London, 4th edition, 1989.